

Attorney Docket No. SON-1531

Date: April 1, 1999

ASSISTANT COMMISSIONER FOR PATENTS  
Washington, D.C. 20231

Sir:

Transmitted herewith for filing is the patent application of

Inventor: Yoshikazu KUROSE

For: IMAGE PROCESSING APPARATUS AND METHOD OF THE SAME

Enclosed are:

- ☒ Specification and Claim(s).
- ☒ Oath or Declaration (unexecuted).
- ☒ TWELVE sheet(s) of drawings.
- ☐ An assignment of the invention to \_\_\_\_\_
- ☐ Copy of \_\_\_\_ Priority application(s).
- ☐ Associate Power of Attorney.

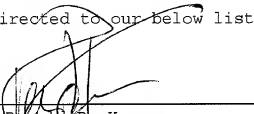
The fee has been calculated as shown below:

CLAIMS AS FILED				
FOR	NUMBER FILED	NUMBER EXTRA	RATE	BASIC FEE
TOTAL CLAIMS	40-20	20	X \$09 \$18	\$380/\$760 \$360
INDEP. CLAIMS	12-3	9	X \$39 \$78	\$702
Fee for Multiple Dependent Claims \$130/\$260				0
			TOTAL FILING FEE	\$1822.00

- ☒ A Letter to Official Draftsperson is attached.
- ☐ A Verified Statement claiming small entity status is enclosed.
- ☒ Charge \$ 1,822.00 to Deposit Account No. 18-0013 to cover the filing fee. A duplicate copy of this sheet is enclosed.
- ☒ The Commissioner is hereby authorized to charge any fees under 37 C.F.R. 1.16 or 1.17 which may be required during the entire pendency of this application, or to credit any overpayment, to Deposit Account No. 18-0013. A duplicate copy of this sheet is enclosed.
- ☐ A check in the amount of \$ \_\_\_\_\_ cover the filing fee is enclosed.
- ☐ Charge \$ \_\_\_\_\_ to Deposit Account No. 18-0013 to cover the recordal fee. A duplicate copy of this sheet is enclosed.
- ☒ Applicant's undersigned attorney may be reached by telephone in our Washington D.C. Office at

(202) 955-3750.

All correspondence should be directed to our below listed address.

  
\_\_\_\_\_  
Ronald P. Kananen  
Reg. No. 24,104

RADER, FISHMAN & GRAUER, P.L.L.C  
1233 20<sup>th</sup> Street, NW, Suite 501  
Washington, DC 20036  
Telephone: (202) 955-3750  
Facsimile: (202) 955-3751

IMAGE PROCESSING APPARATUS AND METHOD OF THE SAME

## BACKGROUND OF THE INVENTION

## 5 1. Field of the Invention

The present invention relates to an image processing apparatus and method which can reduce the power consumption.

## 2. Description of the Related Art

10 Computer graphics are often used in a variety of computer aided design (CAD) systems and amusement machines. Especially, along with the recent advances in image processing techniques, systems using three-dimensional computer graphics are becoming rapidly  
15 widespread.

In three-dimensional computer graphics, the color value of each pixel is calculated at the time of deciding the color of each corresponding pixel. Then, rendering is performed for writing the calculated value  
20 to an address of a display buffer (frame buffer) corresponding to the pixel.

One of the rendering methods is polygon rendering. In this method, a three-dimensional model is expressed as an composite of triangular unit graphics  
25 (polygons). By drawing the polygons as units, the colors

of the pixels of the display screen are decided.

In polygon rendering, coordinates  $(x, y, z)$ , color data  $(R, G, B, \alpha)$ , homogeneous coordinates  $(s, t)$  of texture data indicating a composite image pattern, and  
5 a value of the homogeneous term  $q$  for the respective vertexes of the triangle in a physical coordinate system are input and processing is performed for interpolation of these values inside the triangle.

Here, the homogeneous term  $q$  is, simply stated,  
10 like an expansion or reduction rate. Coordinates in a UV coordinate system of an actual texture buffer, namely, texture coordinate data  $(u, v)$ , are comprised of the homogeneous coordinates  $(s, t)$  divided by the homogeneous term  $q$  to give " $s/q$ " and " $t/q$ " which in turn are  
15 multiplied by texture sizes  $USIZE$  and  $VSIZE$ , respectively.

In the three-dimensional computer graphic system, for example, when drawing in the display buffer (frame buffer), the texture data is read from the texture  
20 buffer by using the texture coordinate data  $(u, v)$  for every pixel and texture mapping is performed for applying the read texture data in units of triangles on the surface of the three-dimensional model.

Note that in texture mapping on a three-  
25 dimensional model, the expansion or reduction rate of the

image indicated by the texture data to be applied to each pixel changes.

In a three-dimensional computer graphic system, however, there are some cases for example where  
5 processing is performed in parallel (simultaneously) for 8 pixels in a predetermined block.

Also, in the above polygon rendering performed in units of triangles, the reduction rate etc. of the texture data to be applied are determined in units of  
10 triangles.

Accordingly, in the results of the 8 pixels' worth of operations processed in parallel, the results of operations on the pixels outside the triangle covered become invalid.

15 Specifically, as shown in Fig. 12, consider the case where a reduction rate is determined by performing a predetermined operation with respect to a triangle 30 and texture mapping is performed by using texture data in accordance with the reduction rate.

20 Here, the blocks 31, 32, and 33 are regions where 8 ( $2 \times 4$ ) bits to be processed in parallel are arranged. In polygon rendering, the same texture data is used for the 8 pixels belonging to one block.

In the case shown in Fig. 12, all of the 8  
25 pixels belonging to the block 32 are located inside the

triangle 30, so the results of the operations on the 8  
pixels are all the valid "1". On the other hand, among  
the 8 pixels belonging to the blocks 31 and 33, 3 pixels  
are inside the triangle 30 and 5 pixels are outside the  
5 triangle 30. Therefore, the results of the operations on  
3 pixels are valid in the 8 pixels, but the results of  
the operations on the 5 pixels become invalid.

In the related art, the polygon rendering was  
performed unconditionally on all of the 8 pixels in the  
10 blocks.

Summarizing the problem to be solved by the  
present invention, as explained above, when performing  
the polygon rendering using triangles as unit graphics,  
if processing is performed on all of the plurality of  
15 pixels located in a block regardless of whether they are  
inside the triangle covered or not, an enormous number of  
invalid operations are performed, so there is a large  
effect on the power consumption.

Also, in a three-dimensional computer graphic  
20 system, unnecessary operations are performed due to a  
variety of reasons in addition to the reason above in  
some cases.

Further, since the clock frequency for  
operation of a three-dimensional computer graphic system  
25 has been becoming very high recently, reduction of the

power consumption has become a significant issue.

#### SUMMARY OF THE INVENTION

An object of the present invention is to provide an  
5 image processing apparatus and method enabling a major  
reduction in the power consumption

To attain the above object, according to a first  
aspect of the present invention, there is provided an  
image processing apparatus comprising a plurality of  
10 pixel processing circuits, each provided for processing  
each of a plurality of pixel data to be processed  
simultaneously, for processing a plurality of input pixel  
data in parallel and a control circuit for stopping the  
operation of the pixel processing circuit when the  
15 processing of the pixel data to be processed in the  
processing circuit is not needed.

According to a second aspect of the present  
invention, there is provided an image processing  
apparatus for expressing an image to be displayed on a  
20 display means by a composite of graphic units of a  
predetermined shape, processing pixel data of a plurality  
of pixels positioned within the same graphic unit on the  
basis of the same processing conditions, and using as  
valid data the results of the processing of the pixel  
25 data of the pixels positioned within the graphic unit to

be processed among pixel data of a plurality of pixels to be processed simultaneously, the image processing apparatus comprising a pixel position judging circuit for judging whether or not a corresponding pixel is  
5 positioned within the graphic unit for each of the plurality of pixel data to be processed simultaneously; a plurality of pixel processing circuits for processing a plurality of pixel data to be processed simultaneously mutually in parallel; and a control circuit for stopping  
10 the operation of the pixel processing circuits other than processing circuits for processing pixel data of pixels positioned within the graphic unit to be processed among the plurality of pixel processing circuits on the basis of the results of the judgement of the pixel position  
15 judging circuit.

According to a third aspect of the present invention, there is provided an image processing apparatus comprising a plurality of image processing  
20 circuits, provided for a plurality of pixels to be processed simultaneously, for blending a plurality of first pixel data and a corresponding plurality of second pixel data by blending ratios indicated by blending ratio data set for each pixel to produce a plurality of third  
25 pixel data and a control circuit for judging whether or not the pixel processing circuits will perform the



blending and stopping the operation of the pixel processing circuits when judging that they will not perform the blending.

According to a fourth aspect of the present invention, there is provided an image processing apparatus for expressing an image to be displayed on a display means by a composite of graphic units of a predetermined shape, processing pixel data of a plurality of pixels positioned within the same graphic unit on the basis of the same processing conditions, and using as valid data the results of the processing of the pixel data of the pixels positioned within the graphic unit to be processed among pixel data of a plurality of pixels to be processed simultaneously, the image processing apparatus comprising a plurality of image processing circuits, provided for a plurality of pixels to be processed simultaneously, for blending a plurality of first pixel data and a corresponding plurality of second pixel data by a blending ratio indicated by blending ratio data set for each pixel to produce a plurality of third pixel data and a control circuit for judging whether or not a corresponding pixel is positioned within a graphic unit for each of the plurality of pixels to be processed simultaneously and stopping the operation of a pixel processing circuit when judging that the

corresponding pixel is not positioned within the graphic unit or when judging that the blending will not be not performed on the basis of the blending ratio data.

According to a fifth aspect of the present  
5 invention, there is provided an image processing apparatus comprising a storage circuit; a plurality of pixel processing circuits, provided for a plurality of pixels to be processed simultaneously, for producing a plurality of second pixel data from a plurality of first  
10 pixel data; a comparing circuit for comparing a plurality of first depth data of the plurality of first pixel data and a plurality of second depth data of a plurality of third pixel data stored in the storage circuit in correspondence with the plurality of first depth data;  
15 and a control circuit for judging whether or not to rewrite third pixel data corresponding to second depth data stored in the storage circuit by second pixel data and stopping the operation of the corresponding pixel processing circuit when judging not to rewrite.

20 According to a sixth aspect of the present invention, there is provided an image processing apparatus for expressing an image to be display on a display means by a composite of graphic units of a predetermined shape, processing pixel data of a plurality  
25 of pixels positioned within the same graphic unit on the

basis of the same processing conditions, and using as valid data the results of the processing of the pixel data of the pixels positioned within the graphic unit to be processed among pixel data of a plurality of pixels to be processed simultaneously, the image processing apparatus comprising a storage circuit; a plurality of pixel processing circuits, provided for a plurality of pixels to be processed simultaneously, for producing a plurality of second pixel data from a plurality of first pixel data; a comparing circuit for comparing a plurality of the first depth data of the plurality of first pixel data and a plurality of second depth data of a plurality of third pixel data stored in the storage circuit in correspondence with the plurality of first depth data; and a control circuit for judging whether or not a corresponding pixel is positioned within the graphic unit for each of the plurality of pixels to be processed simultaneously, judging whether or not to rewrite the third pixel data corresponding to the second depth data stored in the storage circuit with the second pixel data on the basis of the result of the comparison, and stopping the operation of a pixel processing circuit when judging that the corresponding pixel is not positioned within the graphic unit or when judging not to rewrite.

According to a seventh aspect of the present

invention, there is provided an image processing method for performing image processing by using pixel processing circuits, each provided for each of a plurality of pixels to be processed simultaneously, for processing a

- 5 plurality of input pixel data in parallel, comprising the steps of judging whether or not on the basis of the pixel data the pixel processing of the processing circuits is needed, and stopping operation of the pixel processing circuit when judging the pixel processing of the
- 10 processing of the processing circuit is not needed.

According to an eighth aspect of the present invention, there is provided an image processing method for expressing an image to be displayed on a display means by a composite of graphic units of a predetermined

- 15 shape, processing pixel data of a plurality of pixels positioned within the same graphic unit on the basis of the same processing conditions, and using as valid data the results of the processing of the pixel data of the pixels positioned within the graphic unit to be processed
- 20 among pixel data of a plurality of pixels to be processed simultaneously, the image processing method comprising judging whether or not a corresponding pixel is positioned within the graphic unit for each of the plurality of pixel data to be processed simultaneously;
- 25 processing a plurality of pixel data to be processed

simultaneously mutually in parallel in a plurality of pixel processing circuits; and stopping the operation of the pixel processing circuits other than processing circuits for processing pixel data of pixels positioned within the graphic unit to be processed among the plurality of pixel processing circuits on the basis of the results of the judgement.

According to a ninth aspect of the present invention, there is provided an image processing method comprising using a plurality of pixel processing circuits provided for a plurality of pixels to be processed simultaneously to blend a plurality of first pixel data and a plurality of second pixel data by blending ratios indicated by blending ratio data set for each pixel to produce a plurality of third pixel data, judging based on the blending ratio data whether to perform the blending by the pixel processing circuits, and stopping the operation of the corresponding pixel processing circuits when judging that they will not perform the blending.

According to a 10th aspect of the present invention, there is provided an image processing method for expressing an image to be displayed on a display means by a composite of graphic units of a predetermined shape, processing pixel data of a plurality of pixels positioned within the same graphic unit on the basis of the same

processing conditions, and using as valid data the results of the processing of the pixel data of the pixels positioned within the graphic unit to be processed among pixel data of a plurality of pixels to be processed

5 simultaneously, the image processing method comprising using a plurality of image processing circuits, provided for a plurality of pixels to be processed simultaneously, to blend a plurality of first pixel data and a plurality of second pixel data by a blending ratio indicated by

10 blending ratio data set for each pixel to produce a plurality of third pixel data and judging whether or not a corresponding pixel is positioned within a graphic unit for each of the plurality of pixels to be processed simultaneously and stopping the operation of a pixel

15 processing circuit when judging that the corresponding pixel is not positioned within the graphic unit or when judging that the blending will not be not performed on the basis of the blending ratio data.

According to an 11th aspect of the present

20 invention, there is provided an image processing method comprising using a plurality of pixel processing circuits, provided for a plurality of pixels to be processed simultaneously, to produce a plurality of second pixel data from a plurality of first pixel data;

25 comparing a plurality of first depth data of the

plurality of first pixel data and a plurality of second depth data of a plurality of third pixel data stored in a storage circuit in correspondence with the plurality of first depth data; and judging whether or not to rewrite  
5 third pixel data corresponding to second depth data stored in the storage circuit by second pixel data and stopping the operation of the corresponding pixel processing circuit when judging not to rewrite.

According to a 12th aspect of the present invention,  
10 there is provided an image processing method for expressing an image to be display on a display means by a composite of graphic units of a predetermined shape, processing pixel data of a plurality of pixels positioned within the same graphic unit on the basis of the same  
15 processing conditions, and using as valid data the results of the processing of the pixel data of the pixels positioned within the graphic unit to be processed among pixel data of a plurality of pixels to be processed simultaneously, the image processing method comprising  
20 using a plurality of pixel processing circuits, provided for a plurality of pixels to be processed simultaneously, to produce a plurality of second pixel data from a plurality of first pixel data; comparing a plurality of the first depth data of the plurality of first pixel data  
25 and a plurality of second depth data of a plurality of

third pixel data stored in a storage circuit in  
correspondence with the plurality of first depth data;  
and judging whether or not a corresponding pixel is  
positioned within the graphic unit for each of the  
5 plurality of pixels to be processed simultaneously,  
judging whether or not to rewrite the third pixel data  
corresponding to the second depth data stored in the  
storage circuit with the second pixel data on the basis  
of the result of the comparison, and stopping the  
10 operation of a pixel processing circuit when judging that  
the corresponding pixel is not positioned within the  
graphic unit or when judging not to rewrite.

#### BRIEF DESCRIPTION OF THE DRAWINGS

15 These and other objects and features of the present  
invention will become clearer from the following  
description of the preferred embodiments given with  
reference to the accompanying drawings, in which:

Fig. 1 is a view of the system configuration of a  
20 three-dimensional computer graphic system according to a  
first embodiment of the present invention;

Fig. 2 is a view for explaining a format of DDA data  
output from a triangle DDA circuit in Fig. 1;

Fig. 3 is a partial view of the configuration of a  
25 texture engine circuit and a memory I/F circuit shown in



Fig. 1;

Fig. 4 is a view of the configuration inside an operation sub-block shown in Fig. 3;

Fig. 5 is a view of the system configuration of a  
5 three-dimensional computer graphic system according to a second embodiment of the present invention;

Fig. 6 is a partial view of the configuration of a texture engine circuit and a memory I/F circuit shown in Fig. 5;

10 Fig. 7 is a view of the system configuration of a three-dimensional computer graphic system according to a third embodiment of the present invention;

Fig. 8 is a partial view of the configuration of a texture engine circuit and a memory I/F circuit shown in  
15 Fig. 7;

Fig. 9 is a view of the configuration of a modification of the three-dimensional computer graphic system shown in Fig. 5;

Fig. 10 is a view of the configuration of a  
20 modification of the three-dimensional computer graphic system shown in Fig. 7;

Fig. 11 is a view of the configuration of an operation block wherein a clock-enabler in the three-dimensional computer graphic system shown in Fig. 1 is applied and pipeline processing is not performed; and  
25

Fig. 12 is a view for explaining disadvantages of the related art.

#### DESCRIPTION OF THE PREFERRED EMBODIMENTS

5 Below, preferred embodiments will be described with reference to the accompanying drawings.

The explanation will be made of a three-dimensional computer graphic system for displaying in a high speed a desired three-dimensional image of any three-dimensional object model on a display, such as a cathode ray tube (CRT), which is applied to home game machines, etc.

##### First Embodiment

Figure 1 is a view of the system configuration of a three-dimensional computer graphic system 1 according to the first embodiment.

In the three-dimensional computer graphic system 1, a three-dimensional model is expressed by a composite of triangular unit graphics (polygons). By drawing the polygons, this system can decide the color of each pixel on the display screen and perform polygon rendering for display on the screen.

In the three-dimensional computer graphic system 1, a three-dimensional object is expressed by using a z-coordinate for indicating the depth in addition to the (x, y) coordinates for indicating positions on a two-

dimensional plane. Any one point of the three dimensional space can be expressed by the three coordinates (x, y, z).

As shown in Fig. 1, in the three-dimensional  
5 computer graphic system 1, a main memory 2, an I/O  
interface circuit 3, a main processor 4, and a rendering  
circuit 5 are connected via a main bus 6.

Below, the operations of the respective components  
will be explained.

10 The main processor 4, for example, in accordance  
with the state of progress in a game, reads the necessary  
graphic data from the main memory 2, performs clipping,  
lighting, geometrical processing, etc. on the graphic  
data and generates polygon rendering data. The main  
15 processor 4 outputs the polygon rendering data S4 to the  
rendering circuit 5 via the main bus 6.

The I/O interface 3 receives as input the polygon  
rendering data from the outside in accordance with need  
and outputs the same to the rendering circuit via the  
20 main bus 6.

Here, the polygon rendering data includes data of  
each of the three vertexes (x, y, z, R, G, B,  $\alpha$ , s, t, q)  
of the polygon.

The (x, y, z) data indicates the three-dimensional  
25 coordinates of a vertex of the polygon, and the (R, G, B)

data indicates the luminance values of red, green, and blue at the three-dimensional coordinates, respectively.

The data  $\alpha$  indicates a coefficient of blending the R, G, B data of a pixel to be drawn and that of a pixel  
5 already stored in the display buffer 21.

Among the (s, t, q) data, the (s, t) indicates homogeneous coordinates of a corresponding texture and the q indicates the homogenous term. Here, the texture sizes USIZE and VSIZE are respectively multiplied with  
10 the "s/q" and "t/q" to obtain coordinate data (u, v) of the texture. The texture coordinate data (u, v) is used for accessing the texture data stored in the texture buffer 20.

Namely, the polygon rendering data indicates  
15 physical coordinate values of the vertexes of a triangle and values of colors of the vertexes, texture, and fogging.

The rendering circuit 5 will be explained in detail below.

20 As shown in Fig. 1, the rendering circuit 5 comprises a digital differential analyzer (DDA) set-up circuit 10, a triangle DDA circuit 11, a texture engine circuit 12, a memory interface (I/F) circuit 13, a cathode ray tube (CRT) controller circuit 14, a random  
25 access memory (RAM) DAC circuit 15, a dynamic random

access memory (DRAM) 16, and a static random access memory (SRAM) 17.

The DRAM 16 functions as a texture buffer 20, a display buffer 21, a z-buffer 22, and a texture CLUT  
5 buffer 23.

#### DDA Set-up Circuit 10

The DDA set-up circuit 10 performs linear interpolation of the values of the vertexes of the triangle on the physical coordinates in a triangle DDA  
10 circuit 11 in its latter part. The DDA set-up circuit 10, prior to obtaining information of the color and depth of the respective pixels inside the triangle, performs a set-up operation for obtaining the sides of the triangle and the difference in a horizontal direction for the data  
15 (z, R, G, B,  $\alpha$ , s, t, q) indicated by the polygon rendering data S4.

Specifically, this set-up operation uses values of the starting point and the ending point and the distance between the two points to calculate the variation of the  
20 value to find when moving by a unit length.

Also, the DDA set-up circuit 10 determines the 1-bit validity instruction data val indicating for each of the 8 bits simultaneously being processed whether it is inside the triangle in question or not. Specifically, the  
25 validity instruction data val is "1" for a pixel inside

the triangle and "0" for a pixel outside the triangle.

The DDA set-up circuit 10 outputs the calculated variational data S10 and the validity instruction data val of the respective pixels to the triangle DDA circuit

5 11.

#### Triangle DDA Circuit 11

The triangle DDA circuit 11 uses the variational data input from the DDA set-up circuit 10 to calculate the linearly interpolated (z, R, G, B,  $\alpha$ , s, t, q) data of the pixels inside the triangle.

The triangle DDA circuit 11 outputs the data (x, y) for each pixel and the (z, R, G, B,  $\alpha$ , s, t, q, val) data for the pixels at the (x, y) coordinates to the texture engine circuit 12 as DDA data (interpolation data) S11.

15 In the first embodiment, the triangle DDA circuit 11 outputs the DDA data S11 of 8 (=2x4) pixels positioned inside a block being processed in parallel to the texture engine circuit 12.

Here, the (z, R, G, B,  $\alpha$ , s, t, q) data of the DDA data S11 comprises 161 bits of data as shown in Fig. 2.

Specifically, each of the R, G, B, and  $\alpha$  data comprises 8 bits, each of the Z, s, t, and q data comprises 32 bits, and the val data comprises one bit.

Note that among the (z, R, G, B,  $\alpha$ , s, t, q, val) data of the 8 bits being processed in parallel, the val

25

data is referred to as val data  $S220_1$  to  $S220_8$  and the  
(z, R, G, B,  $\alpha$ , s, t, q, val) data are referred to as  
operation data  $S221_1$  to  $S221_8$ .

Namely, the triangle DDA circuit 11 outputs the DDA  
5 data  $S11$  composed of (x, y) data, val data  $S220_1$  to  
 $S220_8$ , and the operation data  $S221_1$  to  $S221_8$  for 8 pixels  
to the texture engine circuit 12.

Texture Engine Circuit 12 and Memory I/F Circuit 13

The calculation of "s/q" and "t/q" by the texture  
10 engine circuit 12 using the DDA data  $S11$ , its calculation  
of the texture coordinate data (u, v), and its reading of  
the data (R, G, B,  $\alpha$ ) from the texture buffer 20 and the  
z-comparison and blending by the memory I/F circuit 13  
are performed successively by a pipeline system in the  
15 operation blocks 200, 201, 202, 203, 204, and 205 shown  
in Fig. 3.

Here, the operation blocks 200, 201, 202, 203, 204,  
and 205 respectively include eight operation sub-blocks  
and perform operations on 8 bits in parallel.

20 Here, the texture engine circuit 12 includes the  
operation blocks 200, 201, 202, and 203, while the memory  
I/F circuit 13 includes the operation blocks 204 and 205.

[Operation Block 200]

The operation block 200 uses the (s, t, q) data  
25 included in the DDA data  $S11$  to perform the operation of

dividing the s data by the q data and the operation of dividing the t data by the q data.

The operation block 200 includes eight operation sub-blocks 200<sub>1</sub> to 200<sub>8</sub>, as shown in Fig. 3.

5        Here, the operation sub-block 200<sub>1</sub> receives as input the operation data S221<sub>1</sub> and the val data S220<sub>1</sub>. When the val data S220<sub>1</sub> is "1", that is, indicates the data is valid, it calculates "s/q" and "t/q" and outputs the result of the calculation as the result of division S200<sub>1</sub> to the operation sub-block 201<sub>1</sub> of the operation block 201.

10        When the val data S220<sub>1</sub> is "0", that is, indicates the data is invalid, the operation sub-block 200<sub>1</sub> does not perform the operations and does not output the result of division S200<sub>1</sub>, or outputs a result of division S200<sub>1</sub> indicating a predetermined provisional value to the operation sub-block 201<sub>1</sub> of the operation block 201.

Also, the operation sub-block 200<sub>1</sub> outputs the val data S220<sub>1</sub> to the later operation sub-block 201<sub>1</sub>.

20        Note that the operation sub-blocks 200<sub>2</sub> to 200<sub>8</sub>, respectively perform the same operations as the operation sub-block 200<sub>1</sub> on the corresponding pixels and output the respective results of division S200<sub>2</sub> to S200<sub>8</sub> and the val data S220<sub>2</sub> to S220<sub>8</sub> to the operation sub-blocks 201<sub>2</sub> to 201<sub>8</sub> in the later operation block 201.



Figure 4 is a view of the configuration of the inside of the operation sub-block 200<sub>i</sub>.

Note that all of the operation sub-blocks shown in Fig. 3 basically have the configuration shown in Fig. 4.

5 As shown in Fig. 4, the operation sub-block 200<sub>i</sub> comprises a clock enabler 210<sub>i</sub>, a data flip-flop 222, a processor element 223, and a flag flip-flop 224.

The clock enabler 210<sub>i</sub> receives as input the val data S220<sub>i</sub> at timings based on the system clock signal S225 and detects the level of the val data S220<sub>i</sub>. When  
10 the val data S220<sub>i</sub> is "1", the clock enabler 210<sub>i</sub>, for example, makes the clock signal S210<sub>i</sub> pulse, while when "0", does not make the clock signal S210<sub>i</sub> pulse.

The data flip-flop 222, when detecting a pulse of  
15 the clock signal S210<sub>i</sub>, fetches the operation data S221<sub>i</sub> and outputs it to the processor element 223.

The processor element 223 uses the input operation data S221<sub>i</sub> to perform the above-mentioned division and outputs the result of division S200<sub>i</sub> to the data flip-  
20 flop 222 of the operation sub-block 201<sub>i</sub>.

The flag flip-flop 224 receives the val data S220<sub>i</sub> at timings based on the system clock signal S225 and outputs it to the flag flip-flop 224 of the operation sub-block 201<sub>i</sub> of the later operation block 201.

25 Note that the system clock signal S225 is supplied

to the clock enablers and the flag flip-flops 224 of all of the operation sub-blocks 200<sub>1</sub> to 200<sub>8</sub>, 201<sub>1</sub> to 201<sub>8</sub>, 202<sub>1</sub> to 202<sub>8</sub>, and 204<sub>1</sub> to 204<sub>8</sub> shown in Fig. 3.

Namely, the processing in the operation sub-blocks  
5 200<sub>1</sub> to 200<sub>8</sub>, 201<sub>1</sub> to 201<sub>8</sub>, 202<sub>1</sub> to 202<sub>8</sub>, and 204<sub>1</sub> to 204<sub>8</sub> are carried out synchronously and the eight operation sub-blocks built in the same operation block perform the processing in parallel.

[Operation Block 201]

10 The operation block 201 has the operation sub-blocks 201<sub>1</sub> to 201<sub>8</sub> and multiplies texture sizes USIZE and VSIZE respectively with "s/q" and "t/q" indicated by the results of division S200<sub>1</sub> to S200<sub>8</sub> input from the operation block 200 to generate the texture coordinate  
15 data (u, v).

The operation sub-blocks 201<sub>1</sub> to 201<sub>8</sub> perform operations only when the results of the level detection of the val data S220<sub>1</sub> to S220<sub>8</sub> by the clock enablers 211<sub>1</sub> to 211<sub>8</sub> are "1" and output the texture coordinate data  
20 S201<sub>1</sub> to S201<sub>8</sub> as the results of the operations to the operation sub-blocks 202<sub>1</sub> to 202<sub>8</sub> of the operation block 202.

[Operation Block 202]

The operation block 202 has the operation sub-blocks  
25 202<sub>1</sub> to 202<sub>8</sub>, outputs a reading request including the

texture coordinate data (u, v) generated in the operation block 201 to the SRAM 17 or DRAM 16 via the memory I/F circuit 13, and reads the texture data stored in the SRAM 17 or the texture buffer 20 via the memory I/F circuit 13 to obtain the data S17 (R, G, B,  $\alpha$ ) stored at the texture address corresponding to the (u, v) data.

Note that the texture buffer 20 stores MIPMAP (textures of a plurality of resolutions) and other texture data corresponding to a plurality of reducing rates. Here, which reducing rate of texture data to use is determined in units of the above triangles using a predetermined algorithm.

The SRAM 17 stores a copy of the texture data stored in the texture buffer 20.

The operation sub-blocks 202<sub>1</sub> to 202<sub>8</sub> perform the reading only when the results of the level detection of the val data S220<sub>1</sub> to S220<sub>8</sub> by the clock enablers 212<sub>1</sub> to 212<sub>8</sub> are "1" and output the read (R, G, B,  $\alpha$ ) data S17 as the (R, G, B,  $\alpha$ ) data S202<sub>1</sub> to S202<sub>8</sub> to the operation sub-blocks 203<sub>1</sub> to 203<sub>8</sub> of the operation block 203.

In the case of a full color mode, the texture engine circuit 12 directly uses the (R, G, B,  $\alpha$ ) data read from the texture buffer 20. In the case of an index color mode, the texture engine circuit 12 reads a color look-up table (CLUT), prepared in advance, from the texture CLUT

buffer 23, transfers and stores the same in the built-in SRAM, and uses the color look-up table to obtain the (R, G, B) data corresponding to the color index read from the texture buffer 20.

5 [Operation Block 203]

The operation block 203 has the operation sub-blocks 203<sub>1</sub> to 203<sub>8</sub> and blends the texture data (R, G, B  $\alpha$ ) S202<sub>1</sub> to S202<sub>8</sub> input from the operation block 202 and the (R, G, B) data included in the DDA data S11 from the triangle  
10 DDA circuit 11 by the blending ratio indicated in the  $\alpha$  data (texture  $\alpha$ ) included in the (R, G, B,  $\alpha$ ) data S202<sub>1</sub> to S202<sub>8</sub> to generate (R, G, B) blended data.

Then, the operation block 203 outputs the generated (R, G, B) blended data and the (R, G, B,  $\alpha$ ) data S203<sub>1</sub> to  
15 203<sub>8</sub>, including the  $\alpha$  data included in the corresponding DDA data S11 to the operation block 204.

The operation sub-blocks 203<sub>1</sub> to 203<sub>8</sub> perform the above blending and output the (R, G, B,  $\alpha$ ) data S203<sub>1</sub> to 203<sub>8</sub> only when results of the level detection of the val  
20 data S220<sub>1</sub> to S220<sub>8</sub> by the clock enablers 213<sub>1</sub> to 213<sub>8</sub> are "1".

[Operation Block 204]

The operation block 204 has the operation sub-blocks 204<sub>1</sub> to 204<sub>8</sub> and performs a z-comparison for the input  
25 (R, G, B,  $\alpha$ ) data S203<sub>1</sub> to S203<sub>8</sub> by using the content of

the z-data stored in the z-buffer 22. When the image drawn by the (R, G, B,  $\alpha$ ) data S203<sub>1</sub> to S203<sub>8</sub> is positioned closer to the viewing point than the image drawn in the display buffer 21 the previous time, the operation block 204 updates the z-buffer 22 and outputs the (R, G, B,  $\alpha$ ) data S203<sub>1</sub> to S203<sub>8</sub> as the (R, G, B,  $\alpha$ ) data S204<sub>1</sub> to 204<sub>8</sub> to the operation sub-blocks 205<sub>1</sub> to 205<sub>8</sub> of the operation block 205.

The operation sub-blocks 204<sub>1</sub> to 204<sub>8</sub> perform the above z-comparison and output the (R, G, B,  $\alpha$ ) of the data S204<sub>1</sub> to S204<sub>8</sub> only when the results of the level detection of the val data S220<sub>1</sub> to S220<sub>8</sub> by the clock enablers 214<sub>1</sub> to 214<sub>8</sub> are "1".

[Operation Block 205]

The operation block 205 has the operation sub-blocks 205<sub>1</sub> to 205<sub>8</sub>, blends the (R, G, B,  $\alpha$ ) of the data S204<sub>1</sub> to 204<sub>8</sub> and the (R, G, B) data already stored in the display buffer 21 by the blending ratio indicated in the  $\alpha$  data included in the (R, G, B,  $\alpha$ ) data S204<sub>1</sub> to S204<sub>8</sub>, and writes the blended (R, G, B) data S205<sub>1</sub> to 205<sub>8</sub> in the display buffer 21.

Note that the DRAM 16 is accessed by the memory I/F circuit 13 simultaneously for 16 pixels.

The operation sub-blocks 205<sub>1</sub> to 205<sub>8</sub> perform the above blending and writing to the display buffer 21 only

when the results of the level detection of the val data S220<sub>1</sub> to S220<sub>8</sub> by the clock enablers 215<sub>1</sub> to 215<sub>8</sub> are "1".

#### CRT Controller Circuit 14

The CRT controller circuit 14 generates an address  
5 for display on a not shown CRT in synchronization with  
the given horizontal and vertical synchronization signals  
and outputs a request for reading the display data from  
the display buffer 21 to the memory I/F circuit 13. In  
response to this request, the memory I/F circuit 13 reads  
10 a certain amount of the display data from the display  
buffer 21. The CRT controller 14 has a built-in first-in-  
first-out (FIFO) circuit for storing the display data  
read from the display buffer 21 and outputs the index  
value of RGB to the RAMDAC circuit 15 at certain time  
15 intervals.

#### RAMDAC Circuit 15

The RAMDAC circuit 15 stores the R, G, B data  
corresponding to the respective index values, transfers  
the digital R, G, B data corresponding to the index value  
20 of RGB input from the CRT controller 14, and generates  
the analog RGB data. The RAMDAC circuit 15 outputs the  
generated R, G, B data to the CRT.

The operation of the entire three-dimensional  
computer graphic system 1 will be explained below.

25 Polygon rendering data S4 is output from the main

processor 4 to the DDA set-up circuit 10 via the main bus  
6. Variational data S10 indicating the sides of the  
triangle and the difference in a horizontal direction  
etc. is generated in the DDA set-up circuit 10.

5        This variational data S10 is output to the triangle  
DDA circuit 11. In the triangle DDA circuit 11, the  
linearly interpolated data ( $z$ ,  $R$ ,  $G$ ,  $B$ ,  $\alpha$ ,  $s$ ,  $t$ ,  $q$ ) for  
each pixel inside the triangle is calculated. Then, the  
calculated ( $z$ ,  $R$ ,  $G$ ,  $B$ ,  $\alpha$ ,  $s$ ,  $t$ ,  $q$ ) data and the ( $x$ ,  $y$ )  
10      data of the vertexes of the triangle are output from the  
triangle DDA circuit 11 to the texture engine circuit 12  
as DDA data S11.

Next, the texture engine circuit 12 and memory I/F  
circuit 13 use the DDA data S11 to calculate " $s/q$ " and  
15      " $t/q$ ", calculate the texture coordinate data ( $u$ ,  $v$ ), read  
the ( $R$ ,  $G$ ,  $B$ ,  $\alpha$ ) data as digital data from the texture  
buffer 20, blend it, and write the result to the display  
buffer 21 successively in the operation blocks 200, 201,  
202, 203, 204, and 205 shown in Fig. 3 in a pipeline  
20      format.

The operation of the pipeline processing of the  
texture engine circuit 12 and the memory I/F circuit 13  
shown in Fig. 3 will be explained below.

Here, a case of, for example, simultaneous  
25      processing on 8 pixels in a block 31 shown in Fig. 6 will

be considered. In this case, the val data  $S220_1$ ,  $S220_2$ ,  $S220_3$ ,  $S220_5$ , and  $S220_6$  indicate "0" and the val data  $S220_4$ ,  $S220_7$ , and  $S220_8$  indicate "1".

The val data  $S220_1$  to  $S220_8$  and the operation data  
5  $S221_1$  to  $S221_8$  are input to the corresponding clock  
enablers  $210_1$  to  $210_8$  of the operation sub-blocks  $200_1$  to  
 $200_8$ .

Then in the clock enablers  $210_1$  to  $210_8$ , the levels  
of the respective val data  $S220_1$  to  $S220_8$  are detected.  
10 Specifically, "1" is detected in the clock enablers  $210_4$ ,  
 $210_7$ , and  $210_8$  and "0" is detected in the clock enablers  
 $210_1$ ,  $210_2$ ,  $210_3$ ,  $210_5$ , and  $210_6$ .

As a result, "s/q" and "t/q" are calculated by  
using the operation data  $S221_4$ ,  $S221_7$ , and  $S221_8$  only in  
15 the operation sub-blocks  $200_4$ ,  $200_7$ , and  $200_8$ . The results  
of the division  $S200_4$ ,  $S200_7$ , and  $S200_8$  are output to the  
operation sub-blocks  $201_4$ ,  $201_7$ , and  $201_8$  of the operation  
block 201.

On the other hand, division is not performed in the  
20 operation sub-blocks  $200_1$ ,  $200_2$ ,  $200_3$ ,  $200_5$ , and  $200_6$ .

Further, in synchronization with the output of the  
results of the division  $S200_4$ ,  $S200_7$ , and  $S200_8$ , the val  
data  $S220_1$  to  $S220_8$  are output to the operation sub-  
blocks  $201_1$  to  $201_8$  of the operation block 201.

25 Next, the clock enablers  $210_1$  to  $210_8$  of the



operation sub-blocks  $201_1$  to  $201_8$  detect the levels of the respective val data  $S220_1$  to  $S220_8$ .

Based on the detection results, only the operation sub-blocks  $201_4$ ,  $201_7$ , and  $201_8$  multiply the texture sizes  
5 USIZE and VSIZE with the "s/q" and "t/q" indicated by the results of the division  $S200_4$ ,  $S200_7$ , and  $S200_8$  to generate the texture coordinate data  $S202_4$ ,  $S202_7$ , and  $S202_8$  and output the same to the operation sub-blocks  $202_4$ ,  $202_7$ , and  $202_8$  of the operation block 202.

10 On the other hand, no operation is performed is in the operation sub-blocks  $201_1$ ,  $201_2$ ,  $201_3$ ,  $201_5$ , and  $201_6$ .

In synchronization with the output of the texture coordinate data  $S202_4$ ,  $S202_7$ , and  $S202_8$ , the val data  $S220_1$  to  $S220_8$  are output to the operation sub-blocks  $202_1$   
15 to  $202_8$  of the operation block 202.

Next, the clock enablers  $212_1$  to  $212_8$  of the operation sub-blocks  $202_1$  to  $202_8$  detect the levels of the respective val data  $S220_1$  to  $S220_8$ .

Then, based on the detection results, only the  
20 operation sub-blocks  $202_4$ ,  $202_7$ , and  $202_8$  read the texture data stored in the SRAM 17 or the texture buffer 20 and read the (R, G, B,  $\alpha$ ) data stored at the texture addresses corresponding to the (s, t) data.

The read (R, G, B,  $\alpha$ ) data  $S202_4$ ,  $S202_7$ , and  $S202_8$   
25 are output to the operation sub-blocks  $203_4$ ,  $203_7$ , and

203<sub>8</sub> of the operation block 203.

No read operation is performed in the operation sub-blocks 202<sub>1</sub>, 202<sub>2</sub>, 202<sub>3</sub>, 202<sub>5</sub>, and 202<sub>6</sub>.

In synchronization with the output of the (R, G, B,  $\alpha$ ) data S202<sub>4</sub>, S202<sub>7</sub>, and S202<sub>8</sub>, the val data S220<sub>1</sub> to S220<sub>8</sub> are output to the sub-blocks 203<sub>1</sub> to 203<sub>8</sub> of the operation block 203.

Next, the clock enablers 212<sub>1</sub> to 212<sub>8</sub> of the operation sub-blocks 203<sub>1</sub> to 203<sub>8</sub> detect the levels of the respective val data S220<sub>1</sub> to S220<sub>8</sub>.

Then, based on the detection results, only the operation sub-blocks 203<sub>4</sub>, 203<sub>7</sub>, and 203<sub>8</sub> blend the texture data (R, G, B,  $\alpha$ ) S202<sub>4</sub>, 202<sub>7</sub>, and 202<sub>8</sub> input from the operation block 202 and the (R, G, B) data included in the DDA data S11 from the triangle DDA circuit 11 by the blending ratio indicated by the  $\alpha$  data (texture  $\alpha$ ) included in the (R, G, B,  $\alpha$ ) data S202<sub>4</sub>, 202<sub>7</sub>, and 202<sub>8</sub> to generate the blended data (R, G, B).

Then, the operation sub-blocks 203<sub>4</sub>, 203<sub>7</sub>, and 203<sub>8</sub> output the generated blended data (R, G, B) and the (R, G, B,  $\alpha$ ) data S203<sub>4</sub>, S203<sub>7</sub>, and S203<sub>8</sub> including the  $\alpha$  data included in the corresponding DDA data S11 to the operation block 204.

On the other hand, no blending is performed in the operation sub-blocks 203<sub>1</sub>, 203<sub>2</sub>, 203<sub>3</sub>, 203<sub>5</sub>, and 203<sub>6</sub>.

Next, the clock enablers  $214_1$  to  $214_8$  of the operation sub-blocks  $204_1$  to  $204_8$  detect the levels of the respective val data  $S220_1$  to  $S220_8$ .

Based on the detection results, only the operation sub-blocks  $204_4$ ,  $204_7$ , and  $204_8$  perform the z-comparison. When the image drawn by the (R, G, B,  $\alpha$ ) data  $S203_4$ ,  $S203_7$ , and  $S203_8$  is positioned closer to the viewing point than the image drawn in the display buffer 21 the previous time, the z-buffer 22 is updated and the (R, G, B,  $\alpha$ ) data  $S203_4$ ,  $S203_7$ , and  $S203_8$  is output as the (R, G, B,  $\alpha$ ) data  $S204_4$ ,  $S204_7$ , and  $S204_8$  to the operation sub-blocks  $205_4$ ,  $205_7$ , and  $205_8$  of the operation block 205.

Next, the clock enablers  $215_1$  to  $215_8$  of the operation sub-blocks  $205_1$  to  $205_8$  detect the levels of the respective val data  $S220_1$  to  $S220_8$ .

Based on the detection results, the (R, G, B) data in the (R, G, B,  $\alpha$ ) data  $S204_4$ ,  $S204_7$ , and  $S204_8$  and the (R, G, B) data already stored in the display buffer 21 are blended by the blending ratio indicated by the  $\alpha$  data. Then, the blended (R, G, B) data  $S205_4$ ,  $S205_7$ , and  $S205_8$  are finally calculated.

Then, the (R, G, B) data  $S205_4$ ,  $S205_7$ , and  $S205_8$  are written in the display buffer 21.

No blending is performed in the operation sub-blocks  $204_1$ ,  $204_2$ ,  $204_3$ ,  $204_5$ , and  $205_6$ .

Namely, the texture engine circuit 12 and the memory I/F circuit 13 do not perform processing on the pixels outside the triangle 30 when simultaneously performing the processing on the pixels inside the block 31 shown in Fig. 6. That is, during the processing on the pixels inside the block 31, the operation sub-blocks 200<sub>1</sub>, 200<sub>2</sub>, 200<sub>3</sub>, 200<sub>4</sub>, 200<sub>5</sub>, 201<sub>1</sub>, 201<sub>2</sub>, 201<sub>3</sub>, 201<sub>4</sub>, 201<sub>5</sub>, 201<sub>6</sub>, 202<sub>1</sub>, 202<sub>2</sub>, 202<sub>3</sub>, 202<sub>4</sub>, 202<sub>5</sub>, 204<sub>1</sub>, 204<sub>2</sub>, 204<sub>3</sub>, 204<sub>4</sub>, 204<sub>5</sub>, 204<sub>6</sub>, 205<sub>1</sub>, 205<sub>2</sub>, 205<sub>3</sub>, 205<sub>4</sub>, and 205<sub>5</sub> are stopped, therefore these operation sub-blocks do not consume any power.

As explained above, according to this three-dimensional computer graphic system 1, it is possible not to perform any operation on the pixels outside a triangle being processed among the 8 pixels being simultaneously processed in the pipeline processing in the texture engine circuit 12.

Therefore, the power consumption in the texture engine circuit 12 can be reduced by a large extent. As a result, a simple and inexpensive power source can be used for the three-dimensional computer graphic system 1.

Note that the texture engine circuit 12 realizes the above functions by installing the clock enabler and a 1-bit flag flip-flop in each operation sub-block, as shown in Figs. 3 and 4. However, since the sizes of the circuit of the clock enabler and the 1-bit flag flip-flop are

small, the size of the texture engine circuit 12 is not increased much.

### Second Embodiment

Figure 5 is a view of the system configuration of a  
5 three-dimensional computer graphic system 451 of the second embodiment.

The three-dimensional computer graphic system 451 is the same as the above three-dimensional computer graphic system 1 except that it judges whether or not to perform  
10 the  $\alpha$  blending for each of the pixels in advance and that it stops the processing in the corresponding operation sub-blocks among the operation sub-blocks which perform the  $\alpha$  blending when judged not to perform the  $\alpha$  blending.

Namely, in the second embodiment, the respective  
15 operation sub-blocks stop processing when the corresponding pixel is outside the triangle being processed in the same way as in the first embodiment. Also, the operation sub-block for performing the  $\alpha$  blending among the operation sub-blocks stops processing  
20 when the corresponding pixel is outside the triangle being processed or the  $\alpha$  data of the corresponding pixel is "0".

As shown in Fig. 5, the three-dimensional computer graphic system 451 comprises a main memory 2, an I/O  
25 interface circuit 3, a main processor 4, and a rendering

circuit 425 connected via a main bus 6.

Components in Fig. 5 given the same reference numerals as in Fig. 1 are the same as the components given the same reference numerals explained in the first embodiment.

Namely, the main memory 2, the I/O interface circuit 3, the main processor 4, and the main bus 6 are the same as those explained in the first embodiment.

Also, as shown in Fig. 5, the rendering circuit 425 comprises a DDA set-up circuit 10, a triangle DDA circuit 411, a texture engine circuit 12, a memory I/F circuit 413, a CRT controller circuit 14, a RAMDAC circuit 15, a DRAM 16, and an SRAM 17.

Here, the DDA set-up circuit 10, the texture engine circuit 12, the CRT controller circuit 14, the RAMDAC circuit 15, the DRAM 16, and the SRAM 17 are the same in those explained in the first embodiment.

Below, the triangle DDA circuit 411 and the memory I/F circuit 413 will be explained.

#### Triangle DDA Circuit 411

The triangle DDA circuit 411 uses the variational data S10 input from the DDA set-up circuit 10 in the same way as in the above first embodiment to calculate the linearly interpolated (z, R, G, B,  $\alpha$ , s, t, q) data of the pixels inside the triangle.

The triangle DDA circuit 411 outputs the (x, y) data of the pixels and the (z, R, G, B,  $\alpha$ , s, t, q, val) data for the pixels at the (x, y) coordinates as DDA data (interpolation data) S11 to the texture engine circuit

5 12.

In the second embodiment, the triangle DDA circuit 411 outputs units of 8 pixels' worth of DDA data S11 of pixels positioned inside the block to be processed in parallel to the texture engine circuit 12.

10 Note that among the data (z, R, G, B,  $\alpha$ , s, t, q, val) of the 8 pixels to be processed in parallel, the val data is referred to as val data S220<sub>1</sub> to S220<sub>8</sub>, and the (z, R, G, B,  $\alpha$ , s, t, q) data is referred to as operation data S221<sub>1</sub> to S221<sub>8</sub>.

15 Namely, the triangle DDA circuit 11 outputs the 8 pixels' worth of DDA data S11 composed of the (x, y) data, the val data S220<sub>1</sub> to S220<sub>8</sub>, and the operation data S221<sub>1</sub> to S221<sub>8</sub> to the texture engine circuit 12.

20 The triangle DDA circuit 411 judges whether or not the  $\alpha$  data in the (z, R, G, B,  $\alpha$ , s, t, q) data generated by linear interpolation as explained above is "0" for the 8 pixels being processed in parallel, that is, judges whether or not to perform the  $\alpha$  blending.

25 Then, the triangle DDA circuit 411 outputs the val data S411a<sub>1</sub> to S411a<sub>8</sub> indicating "0" (not to perform the

$\alpha$  blending) to the memory I/F circuit 413 when the  $\alpha$  data is judged to be "0", while outputs the val data S411a<sub>1</sub> to S411a<sub>8</sub> indicating "1" (to perform the  $\alpha$  blending) to the memory I/F circuit 413 when the  $\alpha$  data is judged to be not "0".

#### Memory I/F Circuit 413

Figure 6 is a view of the configuration of the texture engine circuit 12 and the memory I/F circuit 413.

As shown in Fig. 6, the memory I/F circuit 413 comprises the operation block 204 and the operation block 405.

Note that components in Fig. 6 given the same reference numerals as those in Fig. 3 are the same as the components given the same reference numerals explained in the first embodiment.

Namely, the texture engine circuit 12 is the same as that explained in the first embodiment, and the operation block 204 of the memory I/F circuit 413 is the same as that explained in the first embodiment.

Below, the operation block 405 of the memory I/F circuit 413 will be explained.

#### [Operation Block 405]

The operation block 405 has operation sub-blocks 405<sub>1</sub> to 405<sub>8</sub>, blends the (R, G, B,  $\alpha$ ) data S204<sub>1</sub> to S204<sub>8</sub> input from the operation sub-blocks 204<sub>1</sub> to 204<sub>8</sub> and the



(R, G, B) data already stored in the display buffer 21 by the blending ratio indicated by the  $\alpha$  data included in the respective (R, G, B,  $\alpha$ ) data S204<sub>i</sub> to S204<sub>8</sub>, and writes the blended (R, G, B) data S405<sub>i</sub> to S405<sub>8</sub> to the display buffer 21.

At this time, the operation sub-blocks 405<sub>i</sub> to 405<sub>8</sub> detect the levels of the val data S220<sub>i</sub> to S220<sub>8</sub> respectively from the operation block 204 and the val data S411a<sub>i</sub> to S411a<sub>8</sub> from the triangle DDA circuit 411 shown in Fig. 5 and perform the  $\alpha$  blending only when both of the levels are "1".

Here, the case where both of the levels are "1" means that the pixel is inside the triangle being processed and the  $\alpha$  data of the pixel is not "0" (indicating to perform the  $\alpha$  blending).

Namely, the operation sub-block 405<sub>i</sub> to 405<sub>8</sub> do not perform the  $\alpha$  blending when either of the val data S220<sub>i</sub> to S220<sub>8</sub> or the val data S411a<sub>i</sub> to S411a<sub>8</sub> is "0".

Note that the operation sub-blocks 405<sub>i</sub> to 405<sub>8</sub> write the (R, G, B,  $\alpha$ ) data S204<sub>i</sub> to S204<sub>8</sub> input from the operation sub-blocks 204<sub>i</sub> to 204<sub>8</sub> to the display buffer 21 when the level of the val data S220<sub>i</sub> to S220<sub>8</sub> is "1" and the level of the val data S411a<sub>i</sub> to S411a<sub>8</sub> is "0".

Below, the operation of the three-dimensional computer graphic system 451 will be explained.

The overall operation of the three-dimensional computer graphic system 451 is basically the same as that of the overall operation of the three-dimensional computer graphic system 1 explained in the above first  
5 embodiment.

Also, the operation of the pipeline processing of the texture engine circuit 12 and the memory I/F circuit 413 shown in Fig. 6 is the same as the operation explained in the first embodiment in the case of the  
10 processing in the operation blocks 200 to 204.

Below, the operation of the operation block 405 will be explained.

The (R, G, B,  $\alpha$ ) data S204<sub>1</sub> to S204<sub>8</sub> and the val data S220<sub>1</sub> to S220<sub>8</sub> are output from the operation sub-  
15 blocks 204<sub>1</sub> to 204<sub>8</sub> to the operation sub-blocks 405<sub>1</sub> to 405<sub>8</sub> shown in Fig. 6.

The triangle DDA circuit 411 shown in Fig. 5 judges whether or not the  $\alpha$  data in the (z, R, G, B,  $\alpha$ , s, t, q) data generated by linear interpolation is "0" and outputs  
20 the val data S411a<sub>1</sub> to S411a<sub>8</sub> indicating the result of the judgement to the operation sub-blocks 405<sub>1</sub> to 405<sub>8</sub> shown in Fig. 6.

In the operation sub-blocks 405<sub>1</sub> to 405<sub>8</sub>, the clock enablers 415<sub>1</sub> to 415<sub>8</sub> detect the levels of the val data  
25 S220<sub>1</sub> to S220<sub>8</sub> and S411a<sub>1</sub> to S411a<sub>8</sub>. Only when both of the

levels are "1", is the  $\alpha$  blending performed.

In the  $\alpha$  blending, the (R, G, B,  $\alpha$ ) data S204<sub>1</sub> to S204<sub>8</sub> and the (R, G, B) data already stored in the display buffer 21 are blended by the blending ratio  
5 indicated by the  $\alpha$  data included in the respective (R, G, B,  $\alpha$ ) data S204<sub>1</sub> to S204<sub>8</sub> and the (R, G, B) data S405<sub>1</sub> to S405<sub>8</sub>, is generated. Then, the (R, G, B) data S405<sub>1</sub> to S405<sub>8</sub>, is written in the display buffer 21.

Namely, in the second embodiment, in the operation  
10 sub-blocks 405<sub>1</sub> to 405<sub>8</sub>, the  $\alpha$  blending is not performed when either one of the val data S220<sub>1</sub> to S220<sub>8</sub>, or the val data S411a<sub>1</sub> to S411a<sub>8</sub> is "0".

As explained above, according to the three-dimensional computer graphic system 451, the triangle DDA  
15 circuit 411 judges whether the  $\alpha$  data is "0" or not for every pixel.

Further, in the memory I/F circuit 413, it is possible not to perform the  $\alpha$  blending on pixels with  $\alpha$  data of "0", even if pixels inside the triangle being  
20 processed, among the 8 pixels to be processed in parallel, based on the above result of judgement by the triangle DDA circuit 411.

Therefore, according to the three-dimensional computer graphic system 451, the power consumption can be  
25 further reduced compared with the three-dimensional

computer graphic system 1 of the first embodiment.

Third Embodiment

Figure 7 is a view of the system configuration of a three-dimensional computer graphic system 551 of the third embodiment.

The three-dimensional computer graphic system 551 of the third embodiment, for example, compares the z-data of a pixel to be processed with the corresponding z-data stored in the z-buffer. When the image being drawn this time is positioned further from the viewing point than the image drawn the previous time, the three-dimensional computer graphic system 551 stops the generation of the texture coordinate data (u, v), the reading of the texture data, the texture  $\alpha$  blending, and the  $\alpha$  blending.

As shown in Fig. 7, the three-dimensional computer graphic system 551 comprises a main memory 2, an I/O interface circuit 3, a main processor 4, and a rendering circuit 525 connected via a main bus 6.

Components in Fig. 7 given the same reference numerals as in Fig. 1 are the same as the components given the same reference numerals explained in the first embodiment.

Namely, the main memory 2, the I/O interface circuit 3, the main processor 4, and the main bus 6 are the same as those explained in the first embodiment.

Also, as shown in Fig. 7, the rendering circuit 525 comprises a DDA set-up circuit 10, a triangle DDA circuit 11, a texture engine circuit 512, a memory I/F circuit 513, a CRT controller circuit 14, a RAMDAC circuit 15, a  
5 DRAM 16, and an SRAM 17.

Here, the DDA set-up circuit 10, the triangle DDA circuit 11, the CRT controller circuit 14, the RAMDAC circuit 15, the DRAM 16, and the SRAM 17 are the same as those explained in the first embodiment.

10 Below, the texture engine circuit 512 and the memory I/F circuit 513 will be explained.

Figure 8 is a view of the configuration of the texture engine circuit 512 and the memory I/F circuit 513.

15 As shown in Fig. 8, the texture engine circuit 512 comprises operation blocks 500, 501, 502, 503, and 504.

The memory I/F circuit 513 comprises an operation block 505.

In the third embodiment, the operation blocks 500 to  
20 505 are connected in series in order to simultaneously perform the processing on 8 pixels and pipeline processing.

Here, the operation block 500 performs z-comparison, the operation block 501 calculates the "s/q" and "t/q",  
25 the operation block 502 calculates the texture coordinate

data (u, v), the operation block 503 reads the (R, G, B,  $\alpha$ ) data from the texture buffer 20, the operation block 504 performs the texture  $\alpha$  blending, and the operation block 505 performs the  $\alpha$  blending.

5           [Operation Block 500]

The operation block 500 has operation sub-blocks 500<sub>1</sub> to 500<sub>8</sub> and receives as input the DDA data S11 from the triangle DDA circuit shown in Fig. 7.

10           The operation sub-blocks 500<sub>1</sub> to 500<sub>8</sub> detect the levels of the val data S220<sub>1</sub> to S220<sub>8</sub> included in the DDA data S11 in the respective clock enablers 214<sub>1</sub> to 214<sub>8</sub> and perform the z-comparison when the level is "1" (when the pixel is inside a triangle being processed), while do not perform the z-comparison when the level is not "1".

15           The operation sub-blocks 500<sub>1</sub> to 500<sub>8</sub> compare the z-data of the operation data S221<sub>1</sub> to S221<sub>8</sub> included in the DDA data S11 with the corresponding z-data stored in the z-buffer 22 in the z-comparison.

20           Then, the operation sub-blocks 500<sub>1</sub> to 500<sub>8</sub> output the val data S500a<sub>1</sub> to S500a<sub>8</sub> indicating "1" to the operation sub-blocks 501<sub>1</sub> to 501<sub>8</sub> of the operation block 501 when the image drawn by the operation data S221<sub>1</sub> to S221<sub>8</sub> is positioned closer to the viewing point than the image drawn in the display buffer 21 the previous time

25           and update the corresponding z-data stored in the z-

buffer 22 by the z-data of the operation data  $S221_1$  to  $S221_8$ . At this time, the operation sub-blocks  $500_1$  to  $500_8$  further output the operation data  $S221_1$  to  $S221_8$  to the operation sub-blocks  $501_1$  to  $501_8$ .

5        On the other hand, the operation sub-blocks  $500_1$  to  $500_8$  output the val data  $S500a_1$  to  $S500a_8$  indicating "0" to the operation sub-blocks  $501_1$  to  $501_8$  of the operation block 501 when the image drawn by the operation data  $S221_1$  to  $S221_8$  is not positioned closer to the viewing  
10 point than the image drawn on the display buffer the previous time and do not re-write the corresponding z-data stored in the z-buffer 22.

[Operation Block 501]

      The operation block 501 uses the (s, t, q) data  
15 indicated by the DDA data  $S11$  to perform the operation of dividing the s data by the q data and the operation of dividing the t data by the q data.

      The operation block 501 includes eight operation sub-blocks  $501_1$  to  $501_8$  as shown in Fig. 8.

20        Here, the operation sub-block  $501_1$  receives as input the operation data  $S221_1$  and the val data  $S220_1$  and  $S500a_1$ , judges by the clock enablers  $511_1$  to  $511_8$  whether or not both of the val data  $S220_1$  and  $S500a_1$  are "1", that is, the data is valid, and, when it judges both of  
25 the val data to be "1", calculates "s/q" and "t/q" and

outputs the result of division  $S501_1$  to the operation sub-block  $502_1$  of the operation block 502.

When either of the val data  $S220_1$  or  $S500a_1$  is "0", that is, invalid, the operation sub-block  $501_1$  does not  
5 perform any operation and does not output the result of division  $S501_1$  or outputs the result of division  $S501_1$  indicating a predetermined provisional value to the operation sub-block  $502_1$  of the operation block 502.

Note that the operation sub-blocks  $501_2$  to  $501_8$   
10 perform the same operation as in the operation sub-block  $501_1$  on the corresponding pixels and output the respective results of division  $S501_2$  to  $S501_8$  to the operation sub-blocks  $502_2$  to  $502_8$  of the later operation block 502.

15 [Operation Block 502]

The operation block 502 has operation sub-blocks  $502_1$  to  $502_8$  and multiplies the texture sizes  $USIZE$  and  $VSIZ$  with the "s/q" and "t/q" indicated by the results of division  $S501_1$  to  $S501_8$  input from the operation block  
20 501 to generate the texture coordinate data (u, v).

The operation sub-block  $502_1$  detects the levels of the val data  $S220_1$  and  $S500a_1$  in the clock enabler  $512_1$ , performs an operation only when both of the levels are "1", and outputs the texture coordinate data  $S502_1$  as the  
25 respective calculation results to the operation sub-block



503<sub>1</sub> of the operation block 503.

The operation sub-blocks 502<sub>2</sub> to 502<sub>8</sub> process the corresponding data in the same way as in the operation sub block 502<sub>1</sub>.

5 [Operation Block 503]

The operation block 503 has operation sub-blocks 503<sub>1</sub> to 503<sub>8</sub>, outputs a read request including the texture coordinate data (u, v) generated in the operation block 502 to the SRAM 17 or DRAM 16 via the memory I/F circuit 13, and reads the texture data stored in the SRAM 17 or the texture buffer 20 via the memory I/F circuit 13 to obtain the (R, G, B,  $\alpha$ ) data S17 stored in the texture address corresponding to the (u, v) data.

The operation sub-block 503<sub>1</sub> detects the levels of the val data S220<sub>1</sub> and S500a<sub>1</sub> in the clock enabler 513<sub>1</sub> and, only when both of the levels are "1", carries out the read operation and outputs the read (R, G, B,  $\alpha$ ) data S17 as (R, G, B,  $\alpha$ ) data S503<sub>1</sub> to the operation sub-block 504<sub>1</sub> of the operation block 504.

20 The operation sub-blocks 503<sub>2</sub> to 503<sub>8</sub> process the corresponding data in the same way as in the operation sub-block 503<sub>1</sub>.

[Operation Block 504]

The operation block 504 has operation sub-blocks 504<sub>1</sub> to 504<sub>8</sub>, and blends the texture data (R, G, B,  $\alpha$ )

S503<sub>1</sub> to S503<sub>8</sub>, input from the operation block 503 and the (R, G, B) data included in the corresponding DDA data S11 from the triangle DDA circuit 11 by the blending ratio indicated by the  $\alpha$  data (texture  $\alpha$ ) included in the (R, G, B,  $\alpha$ ) data S503<sub>1</sub> to S503<sub>8</sub> to generate the (R, G, B) blend data.

The operation block 504 outputs the generated (R, G, B) blend data and the (R, G, B,  $\alpha$ ) data S504<sub>1</sub> to S504<sub>8</sub>, including the  $\alpha$  data included in the corresponding DDA data S11 to the operation block 505.

The operation sub-blocks 504<sub>1</sub> to 504<sub>8</sub> detect the levels of the val data S220<sub>1</sub> to S220<sub>8</sub> and S500a<sub>1</sub> to S500a<sub>8</sub> respectively by the clock enablers 514<sub>1</sub> to 514<sub>8</sub> and perform the above blending only when both of the levels are "1".

#### [Operation Block 505]

The operation block 505 has operation sub-blocks 505<sub>1</sub> to 505<sub>8</sub>, blends the input (R, G, B,  $\alpha$ ) data S504<sub>1</sub> to S504<sub>8</sub> and the (R, G, B) data already stored in the display buffer 21 by the blending ratio indicated by the  $\alpha$  data included in the respective (R, G, B,  $\alpha$ ) data S504<sub>1</sub> to S504<sub>8</sub>, and writes the (R, G, B) blended data S505<sub>1</sub> to S505<sub>8</sub> to the display buffer 21.

The operation sub-blocks 505<sub>1</sub> to 505<sub>8</sub> detect the levels of the val data S220<sub>1</sub> to S220<sub>8</sub> and S500a<sub>1</sub> to S500a<sub>8</sub>

in the respective clock enablers and, only when both of the levels are "1", perform the above blending and the writing to the display buffer 21

Below, the operation of the pipeline processing of the texture engine circuit 512 and the memory I/F circuit 513 shown in Fig. 8 will be explained.

First, the clock enablers 214<sub>1</sub> to 214<sub>8</sub> of the operation sub-blocks 500<sub>1</sub> to 500<sub>8</sub> detect the levels of the val data S220<sub>1</sub> to S220<sub>8</sub> included in the DDA data S11. When the detected level is "1" (when the pixel is inside the triangle being processed), the z-comparison is performed.

Then, when the image to be drawn by the operation data S221<sub>1</sub> to S221<sub>8</sub> is positioned closer to the viewing point than the image drawn in the display buffer the previous time, the val data S500a<sub>1</sub> to S500a<sub>8</sub> respectively indicating "1" are output to the operation sub-blocks 501<sub>1</sub> to 501<sub>8</sub> of the operation block 501 and the corresponding z-data stored in the z-buffer 22 is updated by the z-data of the respective operation data S221<sub>1</sub> to S221<sub>8</sub>. At this time, the operation data S221<sub>1</sub> to S221<sub>8</sub> are output from the operation sub-blocks 500<sub>1</sub> to 500<sub>8</sub> to the operation sub-blocks 501<sub>1</sub> to 501<sub>8</sub>.

On the other hand, when the levels of the val data S220<sub>1</sub> to S220<sub>8</sub> are not "1", the z-comparison is not

performed and the val data S500a<sub>1</sub> to S500a<sub>8</sub> indicating  
"0" are output to the operation sub-blocks 501<sub>1</sub> to 501<sub>8</sub>  
of the operation block 501. At this time, the  
corresponding z-data stored in the z-buffer 22 is not  
5 updated.

Next, it is judged in the clock enablers 511<sub>1</sub> to  
511<sub>8</sub> of the operation sub-blocks 501<sub>1</sub> to 501<sub>8</sub> if both the  
val data S220<sub>1</sub> and S500a<sub>1</sub> are "1", that is, valid. When  
it is judged that both are "1", "s/q" and "t/q" are  
10 calculated and output as results of division S501<sub>1</sub> to  
S501<sub>8</sub> to the operation sub-blocks 502<sub>1</sub> to 502<sub>8</sub> of the  
operation block 502.

On the other hand, when any of the val data S220<sub>1</sub> to  
S220<sub>8</sub> or S500a<sub>1</sub> to S500a<sub>8</sub> is judged to be "0", that is,  
15 invalid, no operation is performed in the operation sub-  
blocks 501<sub>1</sub> to 501<sub>8</sub>.

Next, in the clock enablers 512<sub>1</sub> to 512<sub>8</sub> of the  
operation sub-blocks 502<sub>1</sub> to 502<sub>8</sub>, the levels of the val  
data S220<sub>1</sub> to S220<sub>8</sub> and S500a<sub>1</sub> to S500a<sub>8</sub> are detected.

20 Only when both of the levels are "1", the operation  
sub-blocks 502<sub>1</sub> to 502<sub>8</sub> multiply the respective texture  
sizes USIZE and VSIZE with the "s/q" and "t/q" indicated  
by the results of division S501<sub>1</sub> to S501<sub>8</sub> input from the  
operation block 501 to generate the texture coordinate  
25 data (u, v). The texture coordinate data (u, v) is output

respectively to the operation sub-blocks  $503_1$  to  $503_8$ .

Next, in the clock enablers  $513_1$  to  $513_8$  of the operation sub-blocks  $503_1$  to  $503_8$ , the levels of the val data  $S220_1$  to  $S220_8$  and  $S500a_1$  to  $S500a_8$  are detected.

- 5 Only when both of the levels are "1", a read request including the texture coordinate data  $(u, v)$  is output to the SRAM 17, the texture data is read via the memory I/F circuit 13, and the  $(R, G, B, \alpha)$  data  $S17$  stored in the texture address corresponding to the  $(u, v)$  data is
- 10 obtained. The  $(R, G, B, \alpha)$  data  $S17$  is output as the  $(R, G, B, \alpha)$  data  $S503_1$  to  $S503_8$  to the operation sub-blocks  $504_1$  to  $504_8$ .

- Next, the clock enablers  $514_1$  to  $514_8$  of the operation sub-blocks  $504_1$  to  $504_8$  detect the levels of
- 15 the val data  $S220_1$  to  $S220_8$  and  $S500a_1$  to  $S500a_8$ . Then, only when both of the levels are "1", the  $(R, G, B, \alpha)$  data  $S503_1$  to  $S503_8$  and the  $(R, G, B)$  data included in the corresponding DDA data  $S11$  from the triangle DDA circuit 11 are blended by the blending ratio indicated by
- 20 the  $\alpha$  data included in the  $(R, G, B, \alpha)$  data  $S503_1$  to  $S503_8$  to generate the  $(R, G, B)$  blend data.

- Then, the generated  $(R, G, B)$  blend data and the  $(R, G, B, \alpha)$  data  $S504_1$  to  $S504_8$  including the  $\alpha$  data included in the corresponding DDA data  $S11$  are output
- 25 from the operation sub-blocks  $504_1$  to  $504_8$  to the

operation sub-blocks 505<sub>1</sub> to 505<sub>8</sub>.

Next, in the clock enablers 215<sub>1</sub> to 215<sub>8</sub> of the operation sub-blocks 505<sub>1</sub> to 505<sub>8</sub>, the levels of the val data S220<sub>1</sub> to S220<sub>8</sub> and S500a<sub>1</sub> to S500a<sub>8</sub> are detected. Only  
5 when both of the levels are "1", the (R, G, B,  $\alpha$ ) data S504<sub>1</sub> to S504<sub>8</sub> and the (R, G, B) data already stored in the display buffer 21 are blended by the blending ratio indicated by the  $\alpha$  data included in the respective (R, G, B,  $\alpha$ ) data S504<sub>1</sub> to S504<sub>8</sub>. The blended (R, G, B) data  
10 S505<sub>1</sub> to S505<sub>8</sub> are written in the display buffer 21.

As explained above, according to the three-dimensional computer graphic system 551, the first operation block 500 of the texture engine circuit 512 performs the z-comparison on the respective pixels and  
15 judges if the image data to be generated in the latter processing should be written in the display buffer 21 or not.

Then, in the texture engine circuit 512 and memory I/F circuit 513, even a pixel inside the triangle being  
20 processed among the 8 pixels to be processed in parallel is, based on the above results of the judgement by the operation block 500, made to be not processed (stopped) as image data not to be written in the display buffer 21.

Therefore, according to the three-dimensional  
25 computer graphic system 551, the power consumption can be

further reduced compared with the above three-dimensional computer graphic system 1 of the first embodiment.

The present invention is not limited to the above embodiments.

5 For example, in the above second embodiment, as shown in Fig. 6, an example was given of the case where 8 pixels of data were simultaneously processed in the operation blocks of the texture engine circuit 12 and the memory I/F circuit 413, however, 1 pixel of data may be  
10 processed in the operation blocks as well.

In this case, since only the operation data  $S221_i$  of the pixel to be processed is input to the texture engine circuit 12, the val data  $S220_i$  becomes unnecessary. Namely, operations are always performed in the operation  
15 sub-blocks  $200_i$ ,  $201_i$ ,  $202_i$ ,  $203_i$ , and  $204_i$ . In the operation sub-block  $405_i$ , the  $\alpha$  blending is performed only when the level of the val data  $S400a_i$  is "1".

Also, in the above third embodiment, as shown in Fig. 8, an example was given of the case where 8 pixels  
20 of data were simultaneously processed in the operation blocks of the texture engine circuit 512 and the memory I/F circuit 513, however, as shown in Fig. 10, 1 pixel of data may also be processed in the operation blocks.

In this case, since only the operation data  $S221_i$  of  
25 the pixel to be processed is input to the texture engine

circuit 512, the val data  $S220_1$  becomes unnecessary. Namely, the z-comparison is always performed in the operation sub-block  $500_1$ . In the operation sub-blocks  $501_1$ ,  $502_1$ ,  $503_1$ ,  $504_1$ , and  $505_1$ , the processing is  
5 performed only when the level of the val data  $S500a_1$  generated in the operation sub-block  $500_1$  is "1".

Also, in the above embodiments, as shown in Fig. 3, an example was given of the case where the val data  $S220_1$  to  $S220_8$  are used for the operation sub-blocks to perform  
10 the pipeline processing in the texture engine circuit 12 and memory I/F circuit 13. However, for example, whether or not to perform the operation may be determined by using the val data  $S320_1$  to  $S320_8$  as shown in Fig. 11 on a predetermined processing without pipeline processing  
15 among the processing in the DDA set-up-circuit 10, the triangle DDA circuit 11, the texture engine circuit 12, and the memory I/F circuit 13 in the rendering circuit 5 shown in Fig. 1.

Also, in the above embodiments, a configuration  
20 using the SRAM 17 was given as an example, however, it may be configured without the SRAM 17.

The texture buffer 20 and the texture CLUT buffer 23 may be provided outside the DRAM 16.

Also, in the above embodiments, a case of displaying  
25 a three-dimensional image was given as an example,



however, the present invention can be applied to a case of displaying a two-dimensional image by simultaneously processing the data of a plurality of pixels.

Also, in the above embodiments, as shown in Fig. 2,  
5 an example of using the DDA data S11 in which the val data was added as valid instruction data to the data (z, R, G, B,  $\alpha$ , s, t, q) to be image processed, however, the (z, R, G, B,  $\alpha$ , s, t, q) data and the val data may be handled as separate independent data.

10 Also, in the above embodiments, an example was given of the case where the geometric processing for generating polygon rendering data was performed in the main processor 4, however, it can be performed in the rendering circuit 5 as well.

15 Furthermore, in the above embodiments, a triangle was given as an example of the unit graphic, however, the shape of the unit graphic is not limited. For example, it may also be a rectangle.

Summarizing the effects of the invention, as  
20 explained above, according to the image processing apparatus and method of the present invention, the power consumption can be reduced by a large extent.

Therefore, according to the image processing apparatus and method of the present invention, a power  
25 source having a small and simple configuration can be

used and the apparatus can be made smaller in size.

While the invention has been described with  
reference to the specific embodiment chosen for purpose  
of illustration, it should be apparent that numerous  
5 modifications could be made thereto by those skilled in  
the art without departing from the basic concept and  
scope of the invention.

100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1020  
1021  
1022  
1023  
1024  
1025  
1026  
1027  
1028  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079  
1080  
1081  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1098  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1130  
1131  
1132  
1133  
1134  
1135  
1136  
1137  
1138  
1139  
1140  
1141  
1142  
1143  
1144  
1145  
1146  
1147  
1148  
1149  
1150  
1151  
1152  
1153  
1154  
1155  
1156  
1157  
1158  
1159  
1160  
1161  
1162  
1163  
1164  
1165  
1166  
1167  
1168  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189  
1190  
1191  
1192  
1193  
1194  
1195  
1196  
1197  
1198  
1199  
1200  
1201  
1202  
1203  
1204  
1205  
1206  
1207  
1208  
1209  
1210  
1211  
1212  
1213  
1214  
1215  
1216  
1217  
1218  
1219  
1220  
1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1240  
1241  
1242  
1243  
1244  
1245  
1246  
1247  
1248  
1249  
1250  
1251  
1252  
1253  
1254  
1255  
1256  
1257  
1258  
1259  
1260  
1261  
1262  
1263  
1264  
1265  
1266  
1267  
1268  
1269  
1270  
1271  
1272  
1273  
1274  
1275  
1276  
1277  
1278  
1279  
1280  
1281  
1282  
1283  
1284  
1285  
1286  
1287  
1288  
1289  
1290  
1291  
1292  
1293  
1294  
1295  
1296  
1297  
1298  
1299  
1300  
1301  
1302  
1303  
1304  
1305  
1306  
1307  
1308  
1309  
1310  
1311  
1312  
1313  
1314  
1315  
1316  
1317  
1318  
1319  
1320  
1321  
1322  
1323  
1324  
1325  
1326  
1327  
1328  
1329  
1330  
1331  
1332  
1333  
1334  
1335  
1336  
1337  
1338  
1339  
1340  
1341  
1342  
1343  
1344  
1345  
1346  
1347  
1348  
1349  
1350  
1351  
1352  
1353  
1354  
1355  
1356  
1357  
1358  
1359  
1360  
1361  
1362  
1363  
1364  
1365  
1366  
1367  
1368  
1369  
1370  
1371  
1372  
1373  
1374  
1375  
1376  
1377  
1378  
1379  
1380  
1381  
1382  
1383  
1384  
1385  
1386  
1387  
1388  
1389  
1390  
1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412  
1413  
1414  
1415  
1416  
1417  
1418  
1419  
1420  
1421  
1422  
1423  
1424  
1425  
1426  
1427  
1428  
1429  
1430  
1431  
1432  
1433  
1434  
1435  
1436  
1437  
1438  
1439  
1440  
1441  
1442  
1443  
1444  
1445  
1446  
1447  
1448  
1449  
1450  
1451  
1452  
1453  
1454  
1455  
1456  
1457  
1458  
1459  
1460  
1461  
1462  
1463  
1464  
1465  
1466  
1467  
1468  
1469  
1470  
1471  
1472  
1473  
1474  
1475  
1476  
1477  
1478  
1479  
1480  
1481  
1482  
1483  
1484  
1485  
1486  
1487  
1488  
1489  
1490  
1491  
1492  
1493  
1494  
1495  
1496  
1497  
1498  
1499  
1500  
1501  
1502  
1503  
1504  
1505  
1506  
1507  
1508  
1509  
1510  
1511  
1512  
1513  
1514  
1515  
1516  
1517  
1518  
1519  
1520  
1521  
1522  
1523  
1524  
1525  
1526  
1527  
1528  
1529  
1530  
1531  
1532  
1533  
1534  
1535  
1536  
1537  
1538  
1539  
1540  
1541  
1542  
1543  
1544  
1545  
1546  
1547  
1548  
1549  
1550  
1551  
1552  
1553  
1554  
1555  
1556  
1557  
1558  
1559  
1560  
1561  
1562  
1563  
1564  
1565  
1566  
1567  
1568  
1569  
1570  
1571  
1572  
1573  
1574  
1575  
1576  
1577  
1578  
1579  
1580  
1581  
1582  
1583  
1584  
1585  
1586  
1587  
1588  
1589  
1590  
1591  
1592  
1593  
1594  
1595  
1596  
1597  
1598  
1599  
1600  
1601  
1602  
1603  
1604  
1605  
1606  
1607  
1608  
1609  
1610  
1611  
1612  
1613  
1614  
1615  
1616  
1617  
1618  
1619  
1620  
1621  
1622  
1623  
1624  
1625  
1626  
1627  
1628  
1629  
1630  
1631  
1632  
1633  
1634  
1635  
1636  
1637  
1638  
1639  
1640  
1641  
1642  
1643  
1644  
1645  
1646  
1647  
1648  
1649  
1650  
1651  
1652  
1653  
1654  
1655  
1656  
1657  
1658  
1659  
1660  
1661  
1662  
1663  
1664  
1665  
1666  
1667  
1668  
1669  
1670  
1671  
1672  
1673  
1674  
1675  
1676  
1677  
1678  
1679  
1680  
1681  
1682  
1683  
1684  
1685  
1686  
1687  
1688  
1689  
1690  
1691  
1692  
1693  
1694  
1695  
1696  
1697  
1698  
1699  
1700  
1701  
1702  
1703  
1704  
1705  
1706  
1707  
1708  
1709  
1710  
1711  
1712  
1713  
1714  
1715  
1716  
1717  
1718  
1719  
1720  
1721  
1722  
1723  
1724  
1725  
1726  
1727  
1728  
1729  
1730  
1731  
1732  
1733  
1734  
1735  
1736  
1737  
1738  
1739  
1740  
1741  
1742  
1743  
1744  
1745  
1746  
1747  
1748  
1749  
1750  
1751  
1752  
1753  
1754  
1755  
1756  
1757  
1758  
1759  
1760  
1761  
1762  
1763  
1764  
1765  
1766  
1767  
1768  
1769  
1770  
1771  
1772  
1773  
1774  
1775  
1776  
1777  
1778  
1779  
1780  
1781  
1782  
1783  
1784  
1785  
1786  
1787  
1788  
1789  
1790  
1791  
1792  
1793  
1794  
1795  
1796  
1797  
1798  
1799  
1800  
1801  
1802  
1803  
1804  
1805  
1806  
1807  
1808  
1809  
1810  
1811  
1812  
1813  
1814  
1815  
1816  
1817  
1818  
1819  
1820  
1821  
1822  
1823  
1824  
1825  
1826  
1827  
1828  
1829  
1830  
1831  
1832  
1833  
1834  
1835  
1836  
1837  
1838  
1839  
1840  
1841  
1842  
1843  
1844  
1845  
1846  
1847  
1848  
1849  
1850  
1851  
1852  
1853  
1854  
1855  
1856  
1857  
1858  
1859  
1860  
1861  
1862  
1863  
1864  
1865  
1866  
1867  
1868  
1869  
1870  
1871  
1872  
1873  
1874  
1875  
1876  
1877  
1878  
1879  
1880  
1881  
1882  
1883  
1884  
1885  
1886  
1887  
1888  
1889  
1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938  
1939  
1940  
1941  
1942  
1943  
1944  
1945  
1946  
1947  
1948  
1949  
1950  
1951  
1952  
1953  
1954  
1955  
1956  
1957  
1958  
1959  
1960  
1961  
1962  
1963  
1964  
1965  
1966  
1967  
1968  
1969  
1970  
1971  
1972  
1973  
1974  
1975  
1976  
1977  
1978  
1979  
1980  
1981  
1982  
1983  
1984  
1985  
1986  
1987  
1988  
1989  
1990  
1991  
1992  
1993  
1994  
1995  
1996  
1997  
1998  
1999  
2000  
2001  
2002  
2003  
2004  
2005  
2006  
2007  
2008  
2009  
2010  
2011  
2012  
2013  
2014  
2015  
2016  
2017  
2018  
2019  
2020  
2021  
2022  
2023  
2024  
2025  
2026  
2027  
2028  
2029  
2030  
2031  
2032  
2033  
2034  
2035  
2036  
2037  
2038  
2039  
2040  
2041  
2042  
2043  
2044  
2045  
2046  
2047  
2048  
2049  
2050  
2051  
2052  
2053  
2054  
2055  
2056  
2057  
2058  
2059  
2060  
2061  
2062  
2063  
2064  
2065  
2066  
2067  
2068  
2069  
2070  
2071  
2072  
2073  
2074  
2075  
2076  
2077  
2078  
2079  
2080  
2081  
2082  
2083  
2084  
2085  
2086  
2087  
2088  
2089  
2090  
2091  
2092  
2093  
2094  
2095  
2096  
2097  
2098  
2099  
2100  
2101  
2102  
2103  
2104  
2105  
2106  
2107  
2108  
2109  
2110  
2111  
2112  
2113  
2114  
2115  
2116  
2117  
2118  
2119  
2120  
2121  
2122  
2123  
2124  
2125  
2126  
2127  
2128  
2129  
2130  
2131  
2132  
2133  
2134  
2135  
2136  
2137  
2138  
2139  
2140  
2141  
2142  
2143  
2144  
2145  
2146  
2147  
2148  
2149  
2150  
2151  
2152  
2153  
2154  
2155  
2156  
2157  
2158  
2159  
2160  
2161  
2162  
2163  
2164  
2165  
2166  
2167  
2168  
2169  
2170  
2171  
2172  
2173  
2174  
2175  
2176  
2177  
2178  
2179  
2180  
2181  
2182  
2183  
2184  
2185  
2186  
2187  
2188  
2189  
2190  
2191  
2192  
2193  
2194  
2195  
2196  
2197  
2198  
2199  
2200  
2201  
2202  
2203  
2204  
2205  
2206  
2207  
2208  
2209  
2210  
2211  
2212  
2213  
2214  
2215  
2216  
2217  
2218  
2219  
2220  
2221  
2222  
2223  
2224  
2225  
2226  
2227  
2228  
2229  
2230  
2231  
2232  
2233  
2234  
2235  
2236  
2237  
2238  
2239  
2240  
2241  
2242  
2243  
2244  
2245  
2246  
2247  
2248  
2249  
2250  
2251  
2252  
2253  
2254  
2255  
2256  
2257  
2258  
2259  
2260  
2261  
2262  
2263  
2264  
2265  
2266  
2267  
2268  
2269  
2270  
2271  
2272  
2273  
2274  
2275  
2276  
2277  
2278  
2279  
2280

What is claimed is:

1. An image processing apparatus comprising:

a plurality of pixel processing circuits, each provided for processing each of a plurality of pixel data to be processed simultaneously, for processing a plurality of input pixel data in parallel; and

a control circuit for stopping the operation of said pixel processing circuit when the processing of said pixel data to be processed in the processing circuit is not needed.

2. An image processing apparatus as set forth in claim 1, wherein

said pixel processing circuit operates on the basis of a clock signal,

said control circuit supplies said pixel processing circuit with said clock signal when judging the pixel processing is needed and stops the supply of said clock signal to said pixel processing circuit when judging the pixel processing is not needed.

3. An image processing apparatus as set forth in claim 2, wherein each of said pixel processing circuits comprises a plurality series connected processing circuits formed as a pipeline circuit.

4. An image processing apparatus as set forth in claim 3, wherein each of said plurality of processing

circuits connected in series within said pixel processing circuit has a flag storage portion, said flag storage portions of said plurality of processing circuits are connected in series to constitute a shift register, and  
5 said shift register is used to control said pipeline processing and the supply of said clock signal.

5. An image processing apparatus as set forth in claim 1, wherein said pixel processing circuit performs processing with respect to pixel data of red (R), green  
10 (G), and blue (B) of a pixel.

6. An image processing apparatus for expressing an image to be displayed on a display means by a composite of graphic units of a predetermined shape, processing pixel data of a plurality of pixels positioned within the  
15 same said graphic unit on the basis of the same processing conditions, and using as valid data the results of the processing of the pixel data of the pixels positioned within said graphic unit to be processed among pixel data of a plurality of pixels to be processed  
20 simultaneously, said image processing apparatus comprising:

a pixel position judging circuit for judging whether or not a corresponding pixel is positioned within said graphic unit for each of the plurality of pixel data  
25 to be processed simultaneously;

a plurality of pixel processing circuits for processing a plurality of pixel data to be processed simultaneously mutually in parallel; and

a control circuit for stopping the operation of  
5 the pixel processing circuits other than processing circuits for processing pixel data of pixels positioned within the graphic unit to be processed among said plurality of pixel processing circuits on the basis of the results of the judgement of said pixel position  
10 judging circuit.

7. An image processing apparatus as set forth in claim 6, wherein

said pixel processing circuit operates on the basis of a clock signal and

15 said control circuit supplies said clock signal to pixel processing circuits processing the pixel data of pixels positioned inside the graphic unit to be processed, and stops the supply of said clock signal to pixel processing circuits processing the pixel data of  
20 pixels not positioned inside the graphic unit to be processed.

8. An image processing apparatus as set forth in claim 7, wherein each of said pixel processing circuits comprises a plurality of processing circuits connected in  
25 series so as to perform pipeline processing.

9. An image processing apparatus as set forth in claim 8, wherein each of said plurality of processing circuits connected in series within said pixel processing circuit has a flag storage portion, said flag storage  
5 portions of said plurality of processing circuits are connected in series to constitute a shift register, and said shift register is used to control said pipeline processing and the supply of said clock signal.

10. An image processing apparatus as set forth in claim 6, wherein:

said pixel position judging circuit adds validity data indicating the result of the judgement to pixel data processed by said pixel processing circuits and

15 said control circuit judges based on the validity data whether to stop the operation of said pixel processing circuits.

11. An image processing apparatus comprising:  
a plurality of image processing circuits,  
20 provided for a plurality of pixels to be processed simultaneously, for blending a plurality of first pixel data and a corresponding plurality of second pixel data by blending ratios indicated by blending ratio data set for each pixel to produce a plurality of third pixel data  
25 and

a control circuit for judging whether or not said pixel processing circuits will perform said blending and stopping the operation of said pixel processing circuits when judging that said blending will not be performed.

12. An image processing apparatus as set forth in claim 11, wherein

each of said pixel processing circuits operates on the basis of a clock signal and

said control circuit supplies said pixel processing circuit with said clock signal when judging that it will perform blending and stops the supply of said clock signal to said pixel processing circuit when judging that it will not perform said blending.

13. An image processing apparatus as set forth in claim 12, wherein each of said pixel processing circuits comprises a plurality of processing circuits connected in series so as to perform pipeline processing.

14. An image processing apparatus as set forth in claim 13, wherein each of said plurality of processing circuits connected in series within said pixel processing circuit has a flag storage portion, said flag storage portions of said plurality of processing circuits are connected in series to constitute a shift register, and said shift register is used to control said pipeline

processing and the supply of said clock signal.

15. An image processing apparatus as set forth in claim 11,

further comprising a storage circuit for  
5 storing said second pixel data, wherein  
said control circuit rewrites the second pixel  
data stored in said storage circuit by said first pixel  
data when judging that blending will not be performed and  
rewrites the second pixel data stored in the  
10 storage circuit by said third pixel data when judging  
that blending will be performed.

16. An image processing apparatus for expressing an  
image to be displayed on a display means by a composite  
of graphic units of a predetermined shape, processing  
15 pixel data of a plurality of pixels positioned within the  
same said graphic unit on the basis of the same  
processing conditions, and using as valid data the  
results of the processing of the pixel data of the pixels  
positioned within said graphic unit to be processed among  
20 pixel data of a plurality of pixels to be processed  
simultaneously, said image processing apparatus  
comprising:

a plurality of image processing circuits,  
provided for a plurality of pixels to be processed  
25 simultaneously, for blending a plurality of first pixel



data and a corresponding plurality of second pixel data by a blending ratio indicated by blending ratio data set for each pixel to produce a plurality of third pixel data and

5 a control circuit for judging whether or not a corresponding pixel is positioned within a graphic unit for each of said plurality of pixels to be processed simultaneously and stopping the operation of a pixel processing circuit when judging that said corresponding  
10 pixel is not positioned within said graphic unit or when judging that said blending will not be performed on the basis of said blending ratio data.

17. An image processing apparatus comprising:

a storage circuit;  
15 a plurality of pixel processing circuits, provided for a plurality of pixels to be processed simultaneously, for producing a plurality of second pixel data from a plurality of first pixel data;

a comparing circuit for comparing a plurality  
20 of first depth data of said plurality of first pixel data and a plurality of second depth data of a plurality of third pixel data stored in said storage circuit in correspondence with said plurality of first depth data; and

25 a control circuit for judging whether or not to

rewrite third pixel data corresponding to second depth data stored in said storage circuit by second pixel data and stopping the operation of the corresponding pixel processing circuit when judging not to rewrite.

- 5           18. An image processing apparatus as set forth in claim 17, wherein

            said pixel processing circuit operates on the basis of a clock signal and

- said control circuit supplies said pixel  
10   processing circuit with said clock signal when judging to rewrite the third pixel data stored in the storage circuit with the second pixel data and stopping the supply of said clock signal to the pixel processing circuit when judging not to rewrite the third pixel data  
15   stored in the storage circuit by the second pixel data.

            19. An image processing apparatus as set forth in claim 18, wherein each of said pixel processing circuits comprises a plurality of processing circuits connected in series so as to perform pipeline processing.

- 20           20. An image processing apparatus as set forth in claim 19, wherein each of said plurality of processing circuits connected in series within said pixel processing circuit has a flag storage portion, said flag storage portions of said plurality of processing circuits are  
25   connected in series to constitute a shift register, and

said shift register is used to control said pipeline processing and the supply of said clock signal.

21. An image processing apparatus for expressing an image to be display on a display means by a composite of graphic units of a predetermined shape, processing pixel data of a plurality of pixels positioned within the same graphic unit on the basis of the same processing conditions, and using as valid data the results of the processing of the pixel data of the pixels positioned within said graphic unit to be processed among pixel data of a plurality of pixels to be processed simultaneously, said image processing apparatus comprising:

a storage circuit;  
a plurality of pixel processing circuits,  
provided for a plurality of pixels to be processed simultaneously, for producing a plurality of second pixel data from a plurality of first pixel data;

a comparing circuit for comparing a plurality of said first depth data of said plurality of first pixel data and a plurality of second depth data of a plurality of third pixel data stored in said storage circuit in correspondence with said plurality of first depth data;  
and

a control circuit for judging whether or not a corresponding pixel is positioned within said graphic

unit for each of said plurality of pixels to be processed simultaneously, judging whether or not to rewrite said third pixel data corresponding to said second depth data stored in said storage circuit with said second pixel data on the basis of the result of the comparison, and stopping the operation of a pixel processing circuit when judging that said corresponding pixel is not positioned within said graphic unit or when judging not to rewrite.

22. An image processing method for performing image processing by using pixel processing circuits, each provided for each of a plurality of pixels to be processed simultaneously, for processing a plurality of input pixel data in parallel, comprising the steps of:

judging whether or not on the basis of said pixel data the pixel processing of said processing circuits is needed, and

stopping operation of said pixel processing circuit when judging the pixel processing of said processing circuit is not needed.

23. An image processing method as set forth in claim 22, further comprising the steps of

supplying said pixel processing circuit with clock signal when judging the pixel processing is needed, and

stopping the supply of said clock signal to

said pixel processing circuit when judging the pixel processing is not needed.

24. An image processing method as set forth in claim 23, wherein each of said pixel processing circuits  
5 performs pipeline processing by a plurality of processing circuits connected in series.

25. An image processing method as set forth in claim 24, wherein each of said plurality of processing  
circuits connected in series within said pixel processing  
10 circuit has a flag storage portion, said flag storage portions of said plurality of processing circuits are connected in series to constitute a shift register, and said shift register is used to control said pipeline processing and the supply of said clock signal.

15 26. An image processing method as set forth in claim 22, wherein said pixel processing is processing with respect to pixel data for deciding output of red (R), green (G), and blue (B) of a pixel.

27. An image processing method for expressing an  
20 image to be displayed on a display means by a composite of graphic units of a predetermined shape, processing pixel data of a plurality of pixels positioned within the same said graphic unit on the basis of the same processing conditions, and using as valid data the  
25 results of the processing of the pixel data of the pixels

positioned within said graphic unit to be processed among pixel data of a plurality of pixels to be processed simultaneously, said image processing method comprising the steps of:

5               judging whether or not a corresponding pixel is positioned within said graphic unit for each of the plurality of pixel data to be processed simultaneously;

                  processing a plurality of pixel data to be processed simultaneously mutually in parallel in a  
10           plurality of pixel processing circuits; and

                  stopping the operation of the pixel processing circuits other than processing circuits for processing pixel data of pixels positioned within the graphic unit to be processed among said plurality of pixel processing  
15           circuits on the basis of the results of the judgement.

28. An image processing method as set forth in claim 27, further comprising the steps of

                  supplying clock signal to pixel processing circuits processing the pixel data of pixels positioned  
20           inside the graphic unit to be processed, and

                  stopping the supply of said clock signal to pixel processing circuits processing the pixel data of pixels not positioned inside the graphic unit to be processed.

25           29. An image processing method as set forth in

claim 28, wherein each of said pixel processing circuits performs pipeline processing by a plurality of processing circuits connected in series.

30. An image processing method as set forth in  
5 claim 29, wherein each of said plurality of processing circuits connected in series within said pixel processing circuit has a flag storage portion, said flag storage portions of said plurality of processing circuits are connected in series to constitute a shift register, and  
10 said shift register is used to control said pipeline processing and the supply of said clock signal.

31. An image processing method comprising the steps of:

using a plurality of pixel processing circuits  
15 provided for a plurality of pixels to be processed simultaneously to blend a plurality of first pixel data and a plurality of second pixel data by blending ratios indicated by blending ratio data set for each pixel to produce a plurality of third pixel data,  
20 judging based on said blending ratio data whether to perform said blending by said pixel processing circuits, and  
stopping the operation of the corresponding pixel processing circuits when judging that they will not  
25 perform said blending.

32. An image processing method as set forth in claim 31, further comprising the steps of  
operating,

supplying a corresponding pixel processing  
5 circuit with clock signal when judging that it will  
perform blending, and

stopping the supply of said clock signal to a  
corresponding pixel processing circuit when judging that  
it will not perform said blending.

10 33. An image processing method as set forth in  
claim 32, wherein each of said pixel processing circuits  
performs pipeline processing by a plurality of processing  
circuits connected in series.

34. An image processing method as set forth in  
15 claim 33, wherein each of said plurality of processing  
circuits connected in series within said pixel processing  
circuit has a flag storage portion, said flag storage  
portions of said plurality of processing circuits are  
connected in series to constitute a shift register, and  
20 said shift register is used to control said pipeline  
processing and the supply of said clock signal.

35. An image processing method for expressing an  
image to be displayed on a display means by a composite  
of graphic units of a predetermined shape, processing  
25 pixel data of a plurality of pixels positioned within the



same said graphic unit on the basis of the same processing conditions, and using as valid data the results of the processing of the pixel data of the pixels positioned within said graphic unit to be processed among  
5 pixel data of a plurality of pixels to be processed simultaneously, said image processing method comprising the steps of:

using a plurality of image processing circuits, provided for a plurality of pixels to be processed  
10 simultaneously, to blend a plurality of first pixel data and a plurality of second pixel data by a blending ratio indicated by blending ratio data set for each pixel to produce a plurality of third pixel data,

judging whether or not a corresponding pixel is  
15 positioned within a graphic unit for each of said plurality of pixels to be processed simultaneously and

stopping the operation of a pixel processing circuit when judging that said corresponding pixel is not positioned within said graphic unit or when judging that  
20 said blending will not be not performed on the basis of said blending ratio data.

36. An image processing method comprising the steps of:

using a plurality of pixel processing circuits,  
25 provided for a plurality of pixels to be processed

simultaneously, to produce a plurality of second pixel data from a plurality of first pixel data;

comparing a plurality of first depth data of said plurality of first pixel data and a plurality of second depth data of a plurality of third pixel data stored in a storage circuit in correspondence with said plurality of first depth data; and

judging whether or not to rewrite third pixel data corresponding to second depth data stored in said storage circuit by second pixel data and stopping the operation of the corresponding pixel processing circuit when judging not to rewrite.

37. An image processing method as set forth in claim 36, further comprising the steps of

supplying said pixel processing circuit with clock signal when judging to rewrite the third pixel data stored in the storage circuit with the second pixel data, and

stopping the supply of said clock signal to the pixel processing circuit when judging not to rewrite the third pixel data stored in the storage circuit by the second pixel data.

38. An image processing method as set forth in claim 37, wherein each of said pixel processing circuits performs pipeline processing by a plurality of processing

circuits connected in series.

39. An image processing method as set forth in claim 37, wherein each of said plurality of processing circuits connected in series within said pixel processing  
5 circuit has a flag storage portion, said flag storage portions of said plurality of processing circuits are connected in series to constitute a shift register, and said shift register is used to control said pipeline processing and the supply of said clock signal.

10 40. An image processing method for expressing an image to be display on a display means by a composite of graphic units of a predetermined shape, processing pixel data of a plurality of pixels positioned within the same graphic unit on the basis of the same processing  
15 conditions, and using as valid data the results of the processing of the pixel data of the pixels positioned within said graphic unit to be processed among pixel data of a plurality of pixels to be processed simultaneously, said image processing method comprising the steps of:

20 using a plurality of pixel processing circuits, provided for a plurality of pixels to be processed simultaneously, to produce a plurality of second pixel data from a plurality of first pixel data;

comparing a plurality of said first depth data  
25 of said plurality of first pixel data and a plurality of

second depth data of a plurality of third pixel data stored in a storage circuit in correspondence with said plurality of first depth data; and

judging whether or not a corresponding pixel is  
5 positioned within said graphic unit for each of said plurality of pixels to be processed simultaneously, judging whether or not to rewrite said third pixel data corresponding to said second depth data stored in said storage circuit with said second pixel data on the basis  
10 of the result of the comparison,

and stopping the operation of a pixel processing circuit when judging that said corresponding pixel is not positioned within said graphic unit or when judging not to rewrite.

IMAGE PROCESSING APPARATUS AND METHOD OF THE SAME

5

ABSTRACT OF THE DISCLOSURE

An image processing apparatus which can reduce the power consumption by a large extent, wherein a predetermined shape to be displayed on a display is expressed by a composite of unit graphics by performing operations on a plurality of pixels simultaneously and by performing processing on valid results of operations for pixels positioned inside a unit graphic being processed. Clock enablers in operation sub-blocks judge the validity of the corresponding val data. Only operation sub-blocks receiving the corresponding val data indicating validity perform operations. Other operation sub-blocks do not perform operations. The operation blocks perform pipeline processing.

FIG. 1

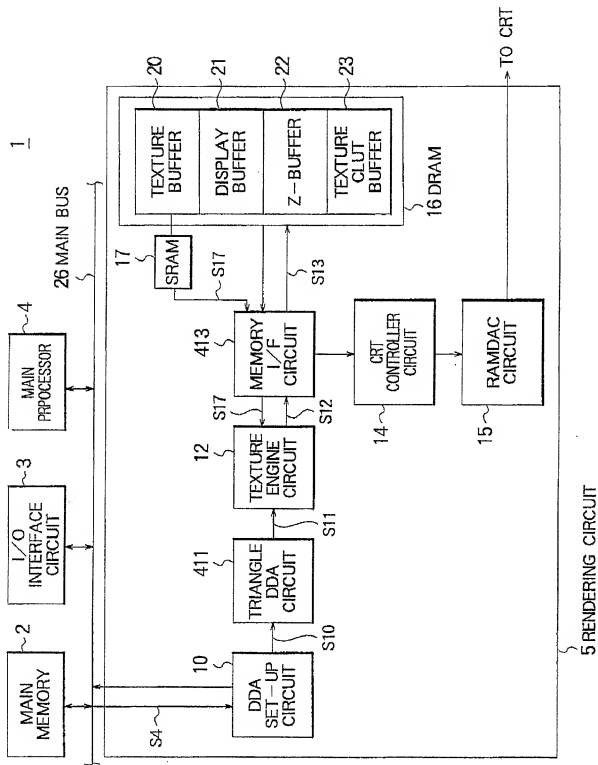


FIG. 2

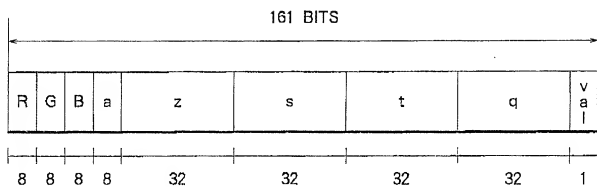


FIG. 3

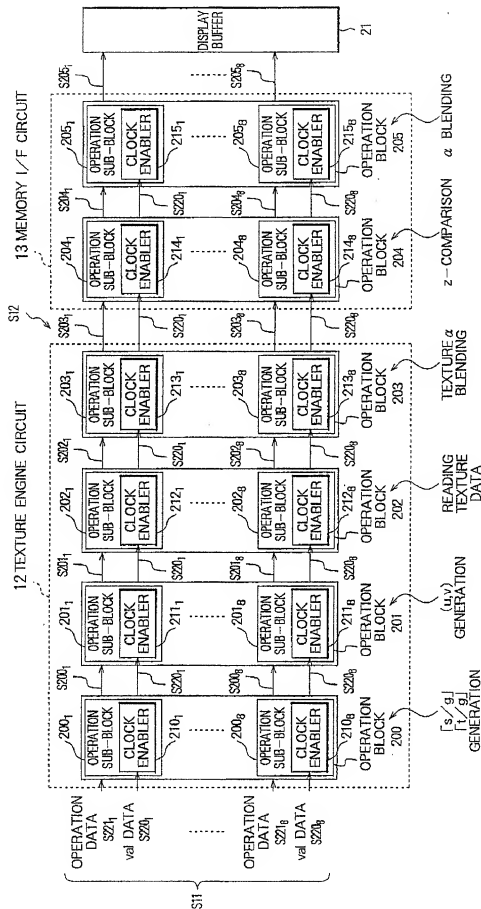






FIG. 5

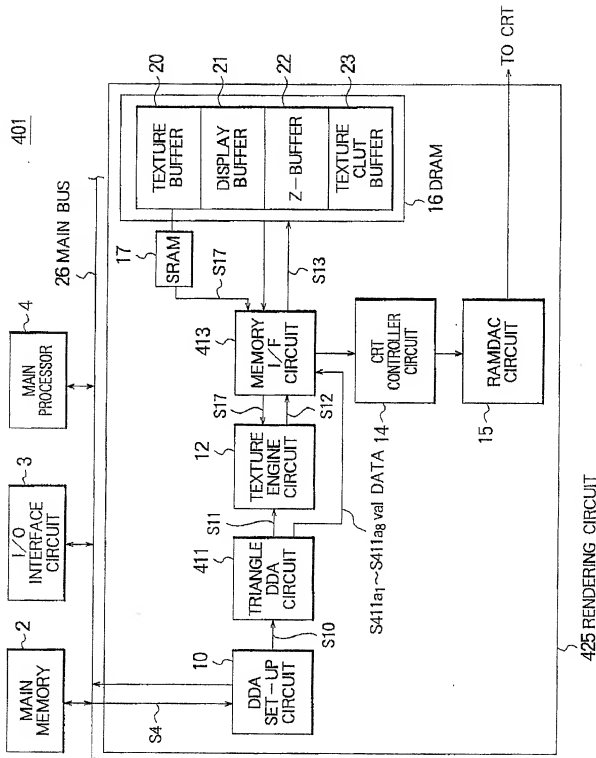


FIG. 6

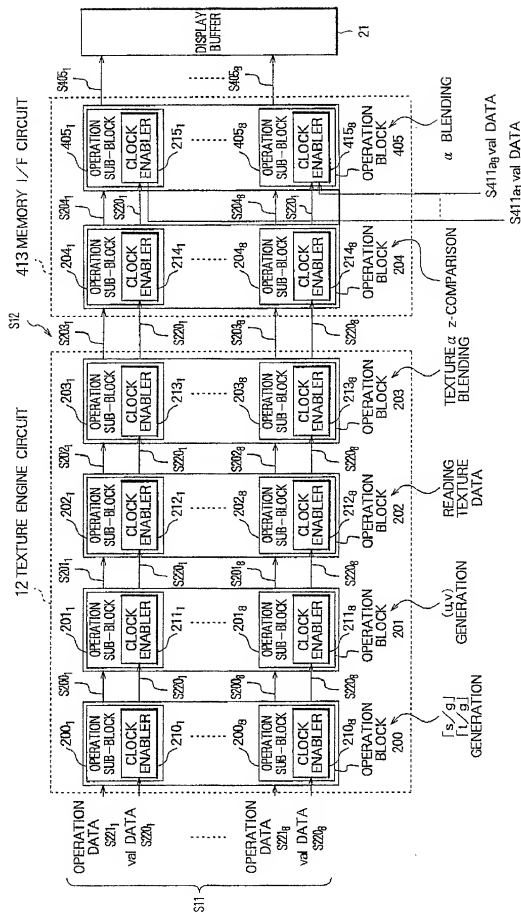


FIG. 7

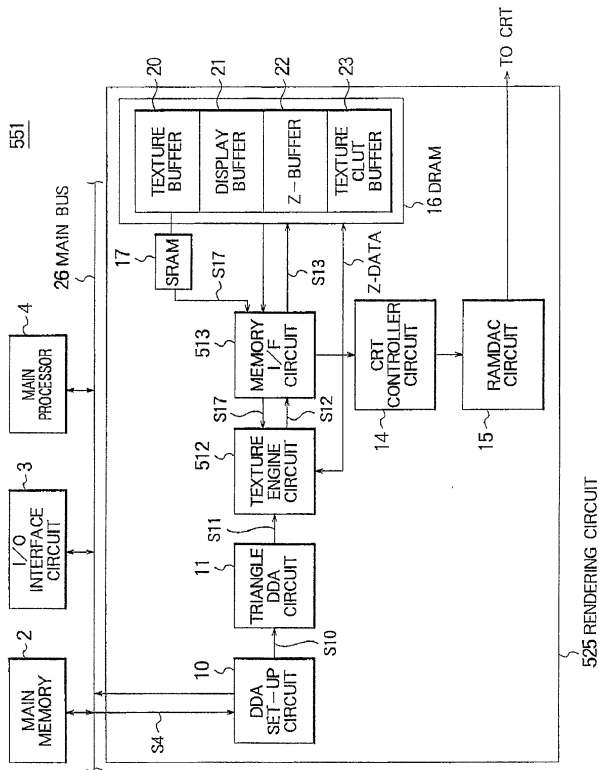




FIG. 9

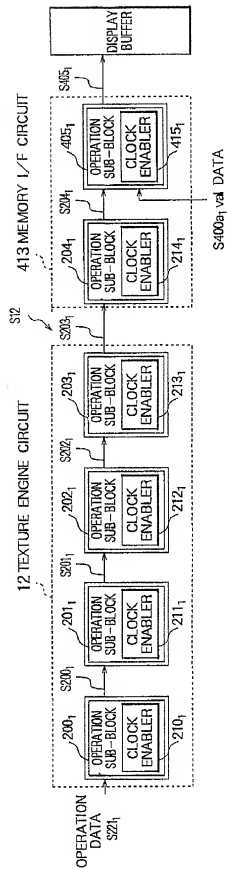


FIG. 10

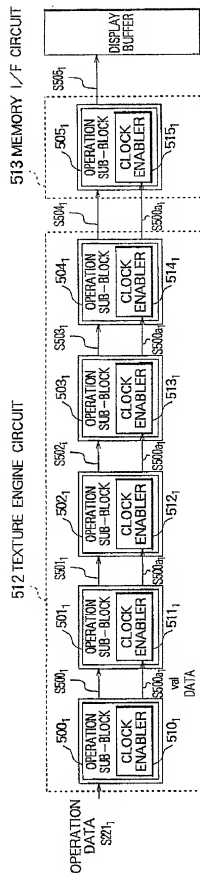


FIG. 11

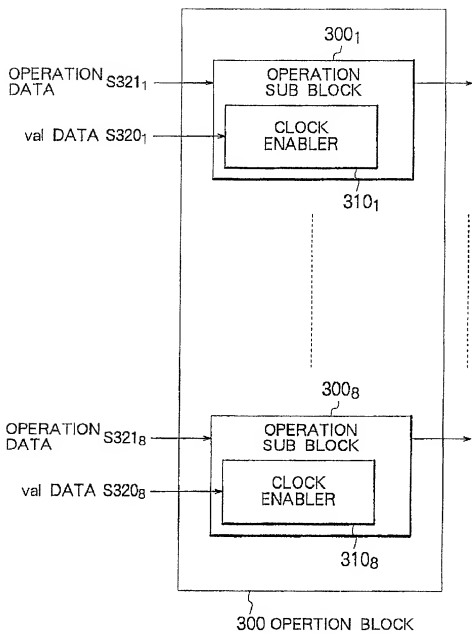
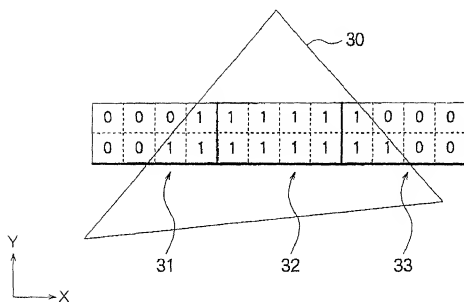




FIG. 12



## DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION

## English Language Declaration

As below named inventors, we hereby declare that:

Our residence, post office address and citizenship are as stated below next to our names.

IMAGE PROCESSING APPARATUS AND METHOD OF THE SAME

We believe we are the original, first and joint inventors of the subject matter which is claimed and for which a patent is sought on the invention entitled

the specification of which

(check one)

☐ is attached hereto.

☐ was filed on \_\_\_\_\_ as

Application Serial No. \_\_\_\_\_

and was amended on \_\_\_\_\_

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent of inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)

Priority Claimed

<u>P10-091844</u>	<u>JAPAN</u>	<u>03/04/1998</u>	X
(Number)	(Country)	(Day/Month/Year Filed)	Yes No
<u>P11-051795</u>	<u>JAPAN</u>	<u>26/02/1999</u>	X
(Number)	(Country)	(Day/Month/Year Filed)	Yes No
_____	_____	_____	Yes No
(Number)	(Country)	(Day/Month/Year Filed)	

We hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code §112, I acknowledge the duty to disclose material to patentability as defined in Title 37, Code of Federal Regulations, §1.56 and 1.63(d) which became available between the filing date of the prior application and the national or PCT international filing date of this application:

\_\_\_\_\_  
(Application Serial No.)

\_\_\_\_\_  
(Filing Date)

\_\_\_\_\_  
(Status)  
(patented, pending, abandoned)

We hereby declare that all statements made herein of our own knowledge are true and that all statements made on information and belief are believed to be true, and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

## English Language Declaration

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

Ronald P. Kananen, Reg. No. 24,104; Ralph T. Rader, Reg. No. 28,772;  
 Michael D. Fishman, Reg. No. 31,951, Richard D. Grauer, Reg. No. 22,388;  
 Joseph V. Coppola, Sr., Reg. No. 33,373; Michael B. Stewart, Reg. No.  
 36,018; Steven L. Nichols, Reg. No. 40,326

Send Correspondence to:

Direct telephone calls to:

Ronald P. Kananen, Esq.  
 RADER, FISHMAN & GRAUER  
 The Lion Building  
 1233 20<sup>th</sup> Street, N.W., Suite 501  
 Washington, D.C. 20036

Ronald P. Kananen, Esq.  
 (202) 955-3750

Full name of first joint inventor YOSHIKAZU KUROSE	
Inventor's signature	Date
Residence Tokyo, JAPAN	
Citizenship JAPAN	
Post Office Address c/o Sony Corporation 7-35, Kitashinagawa, 6-Chome	
Shinagawa-Ku, Tokyo, JAPAN	
Full name of second joint inventor	
Second Inventor's signature	Date
Residence	
Citizenship	
Post Office Address	
Full name of third joint inventor	
Third Inventor's signature	Date
Residence	
Citizenship	
Post Office Address	

(Supply similar information and signature for subsequent joint inventors.)